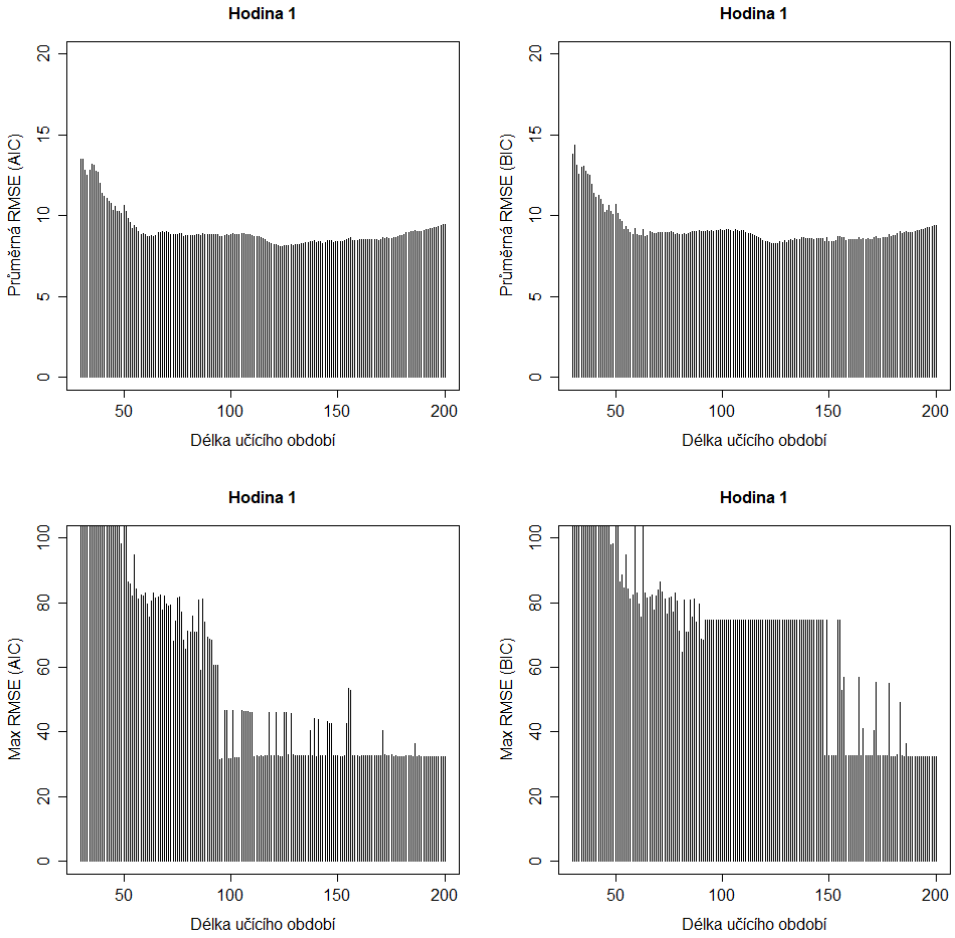
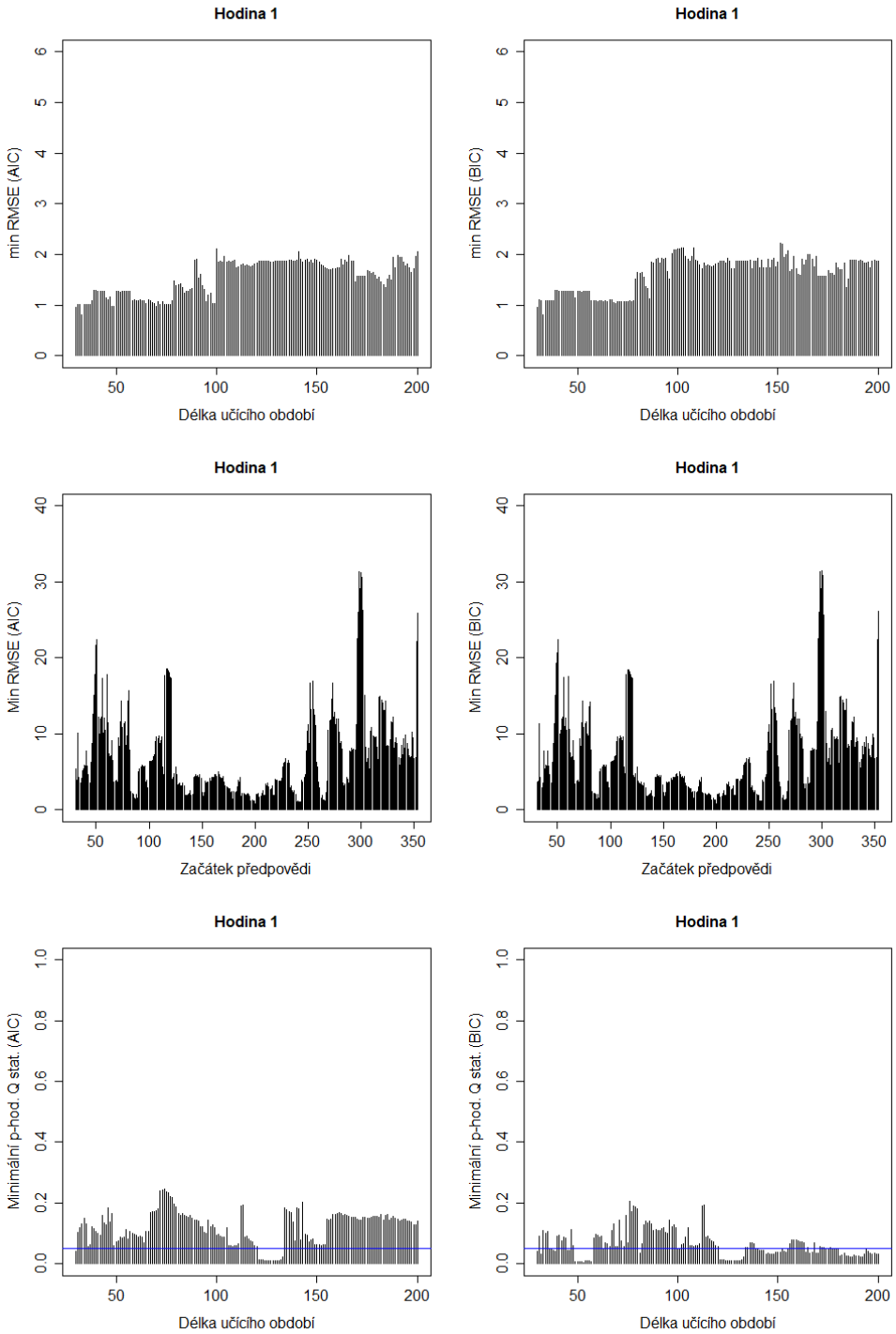
Přílohy

Přílohy

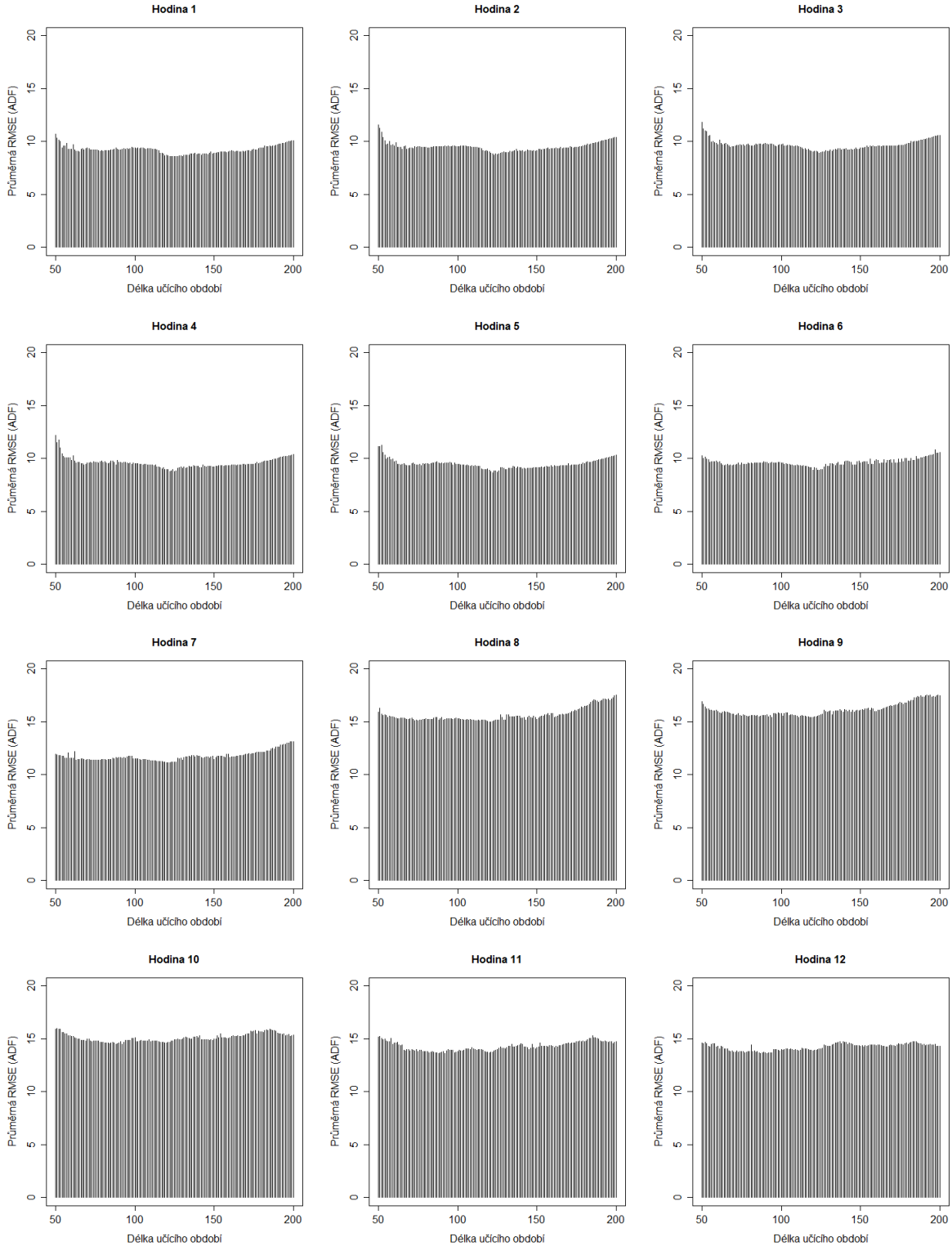
Příloha 1: Sledované statistiky ilustrované na souboru dat prvních hodin

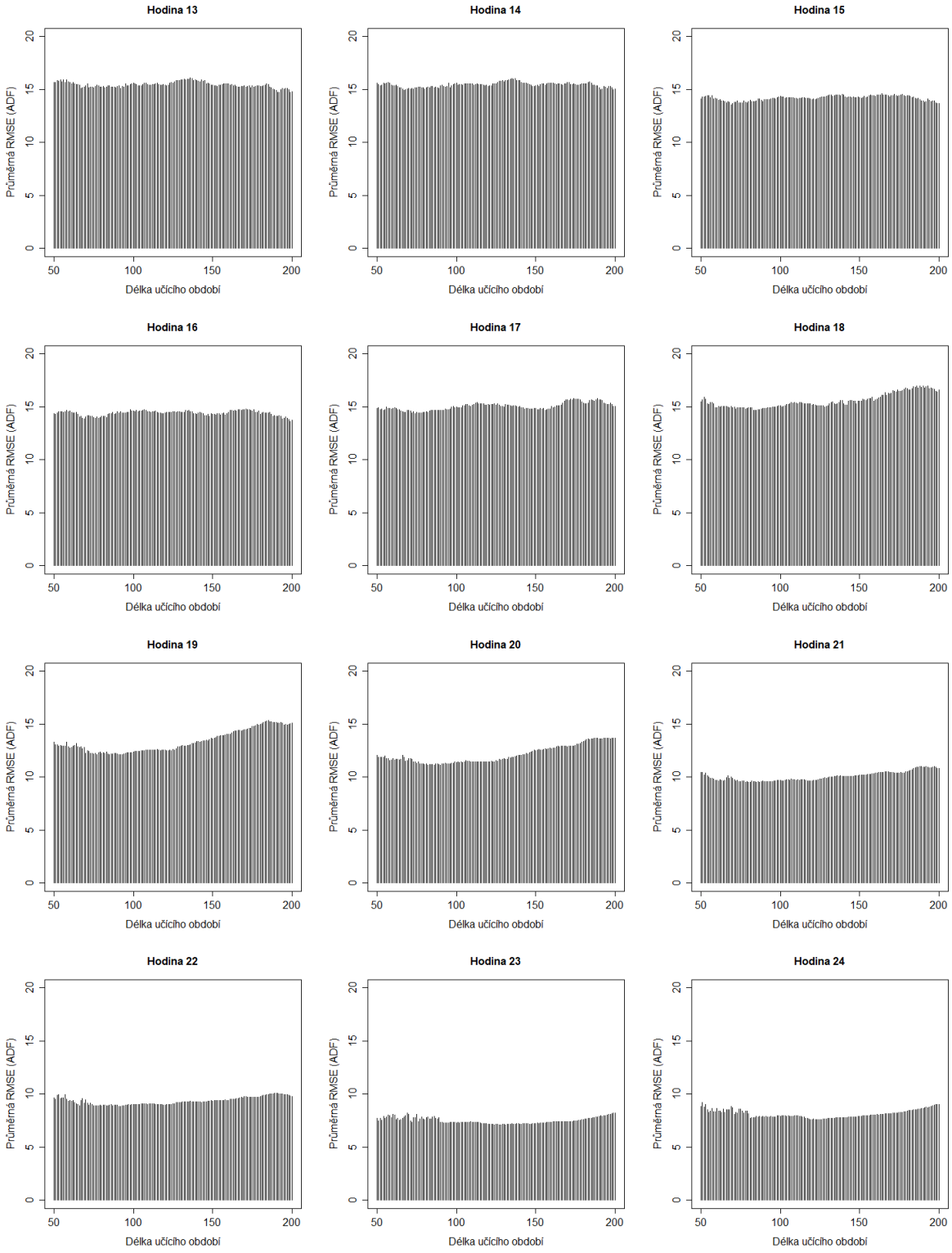




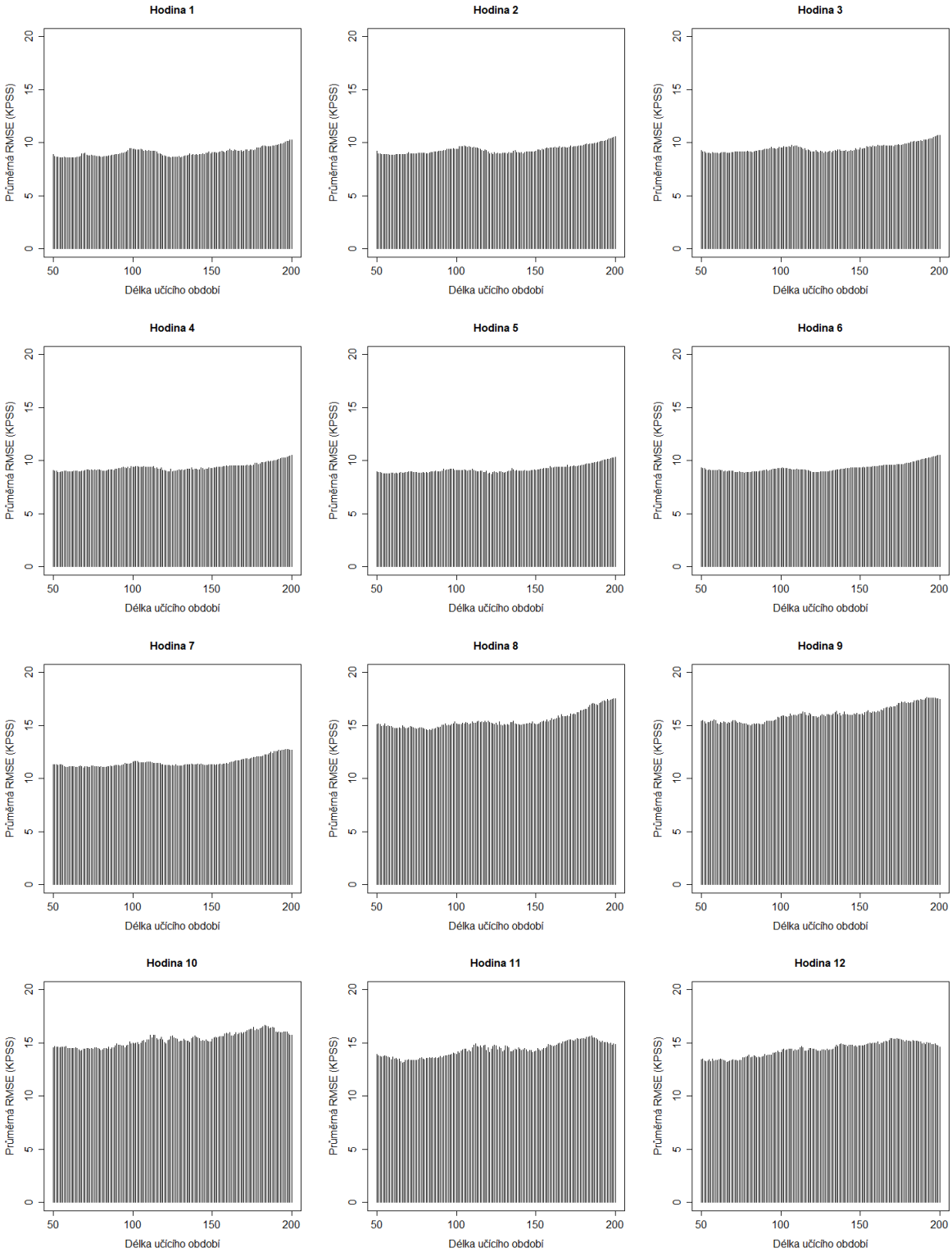
Obr. 1: Průměrná $RMSE$ v závislosti na délce učícího období (1. řádek), maximální $RMSE$ v závislosti na délce učícího období (2. řádek), minimální $RMSE$ v závislosti na délce učícího období (3. řádek), minimální $RMSE$ v závislosti na počátečním dni předpovědi (4. řádek), minimální p–hodnota Q–testu reziduí předpovědního modelu v závislosti na délce učícího období (5. řádek).

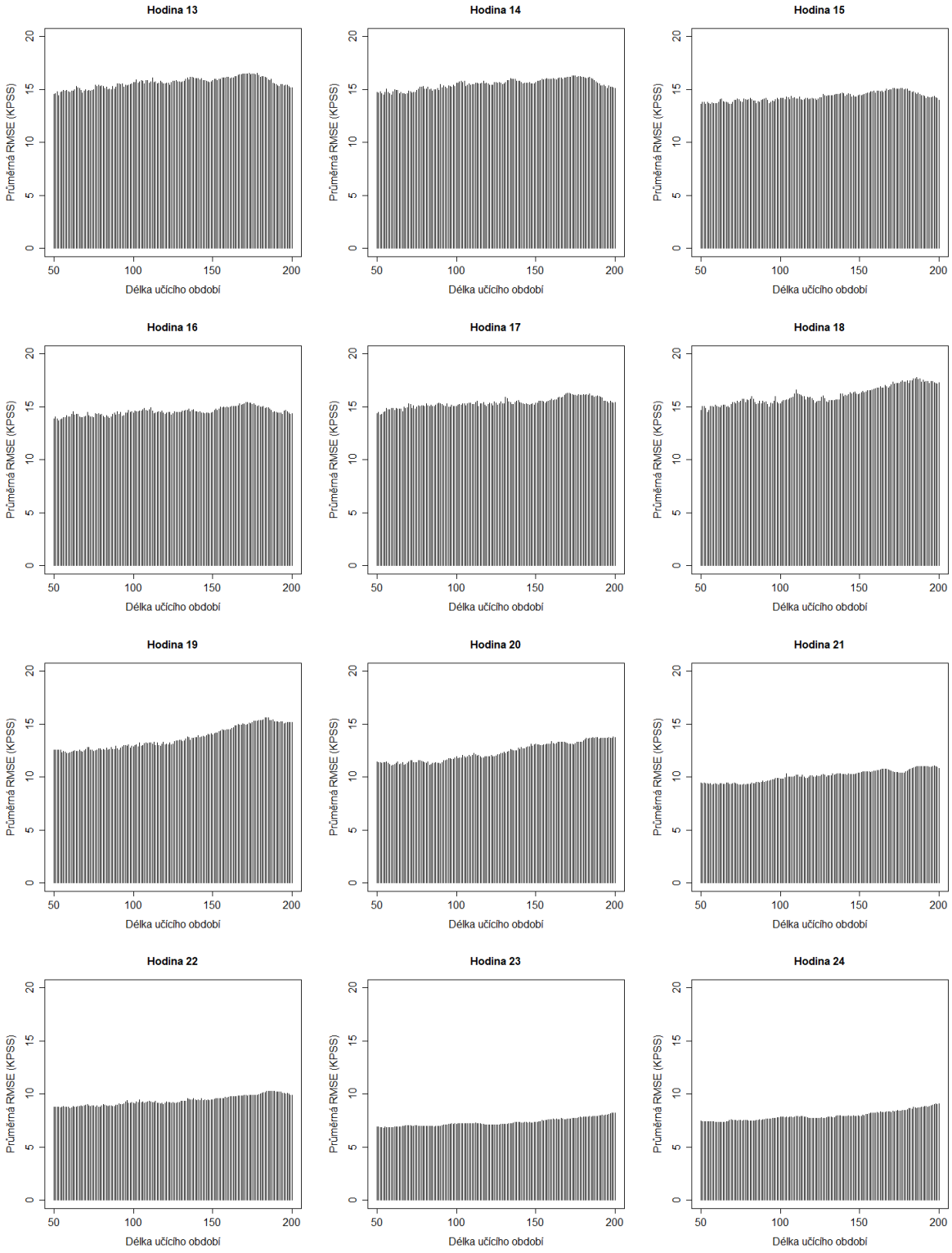
Příloha 2: Statistiky výzkumu pro 24 souborů dat



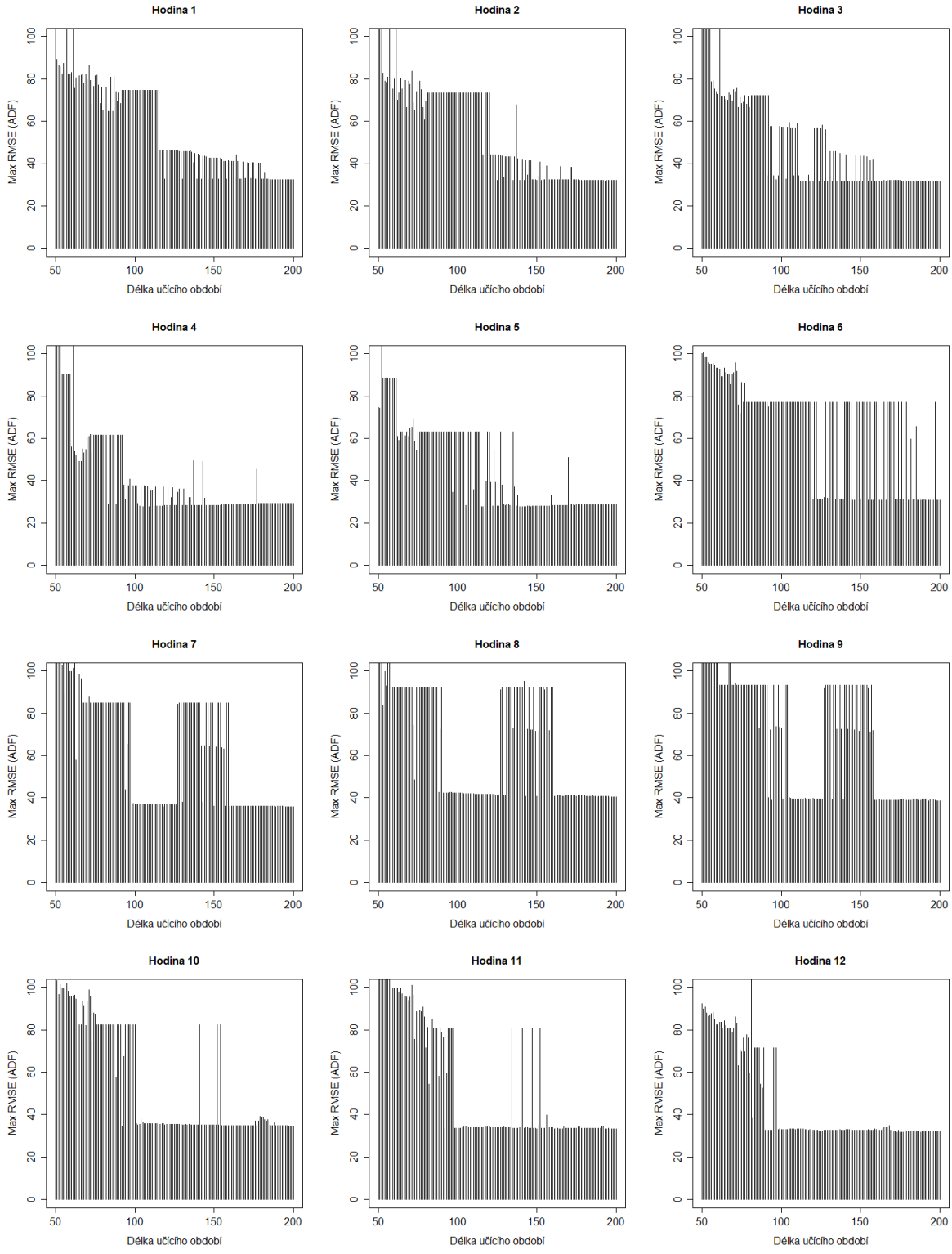


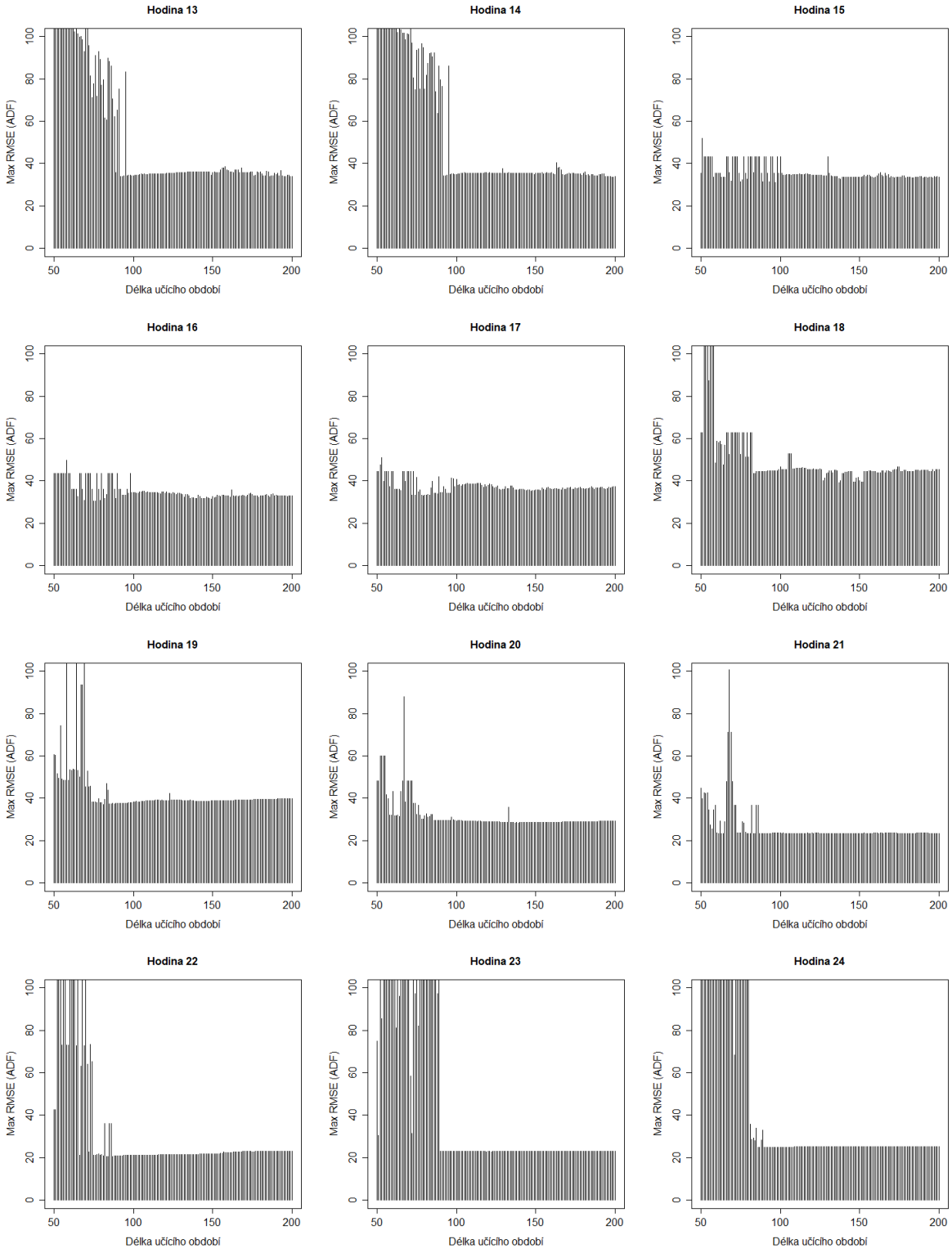
Obr. 2: Průměrná $RMSE$ v závislosti na délce učícího období pro ADF–test.


Hodina 1
Hodina 2
Hodina 3
Hodina 4
Hodina 5
Hodina 6
Hodina 7
Hodina 8
Hodina 9
Hodina 10
Hodina 11
Hodina 12
Průměrná RMSE (KPSS)
Délka učícího období
0
5
10
15
20
50
100
150
200

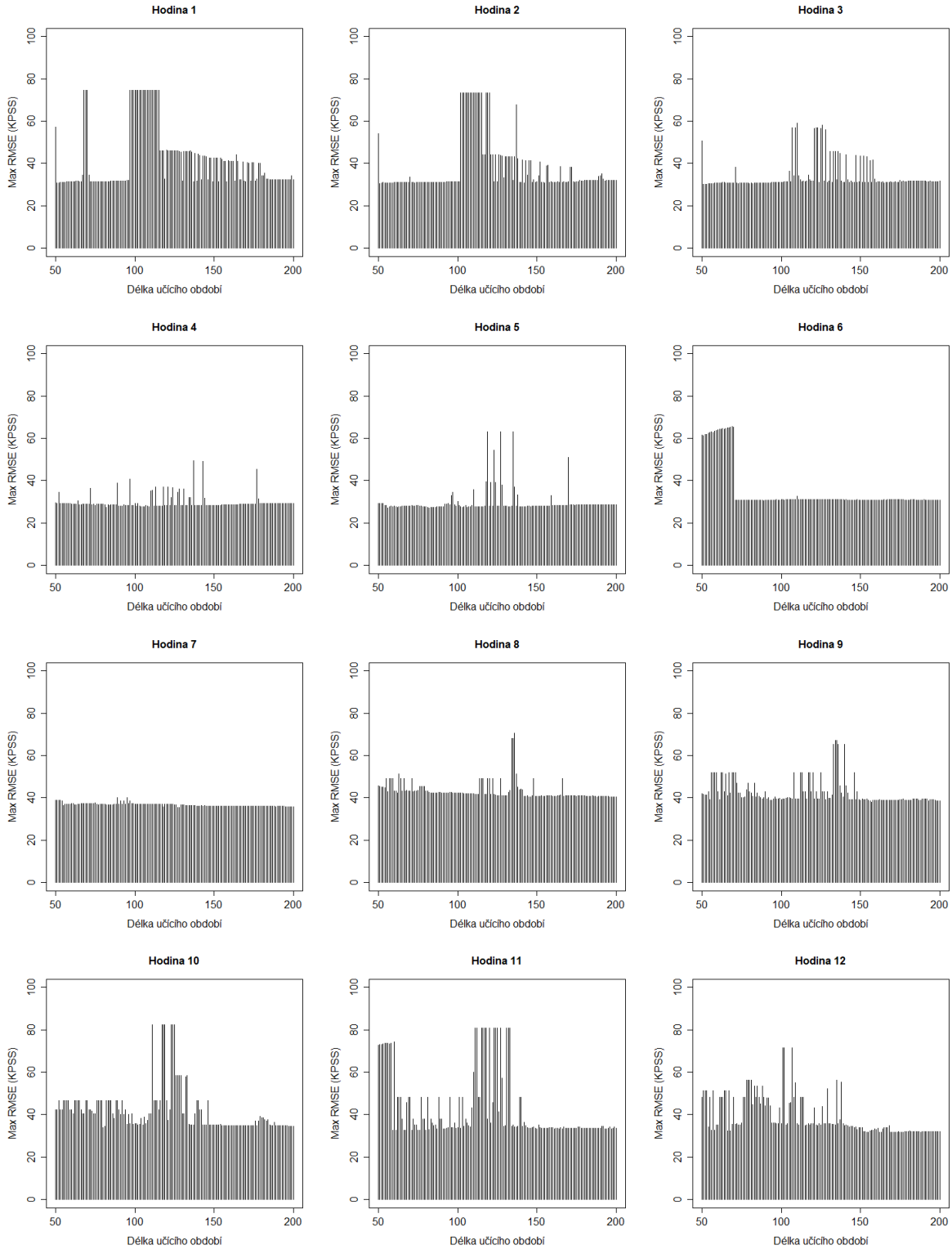


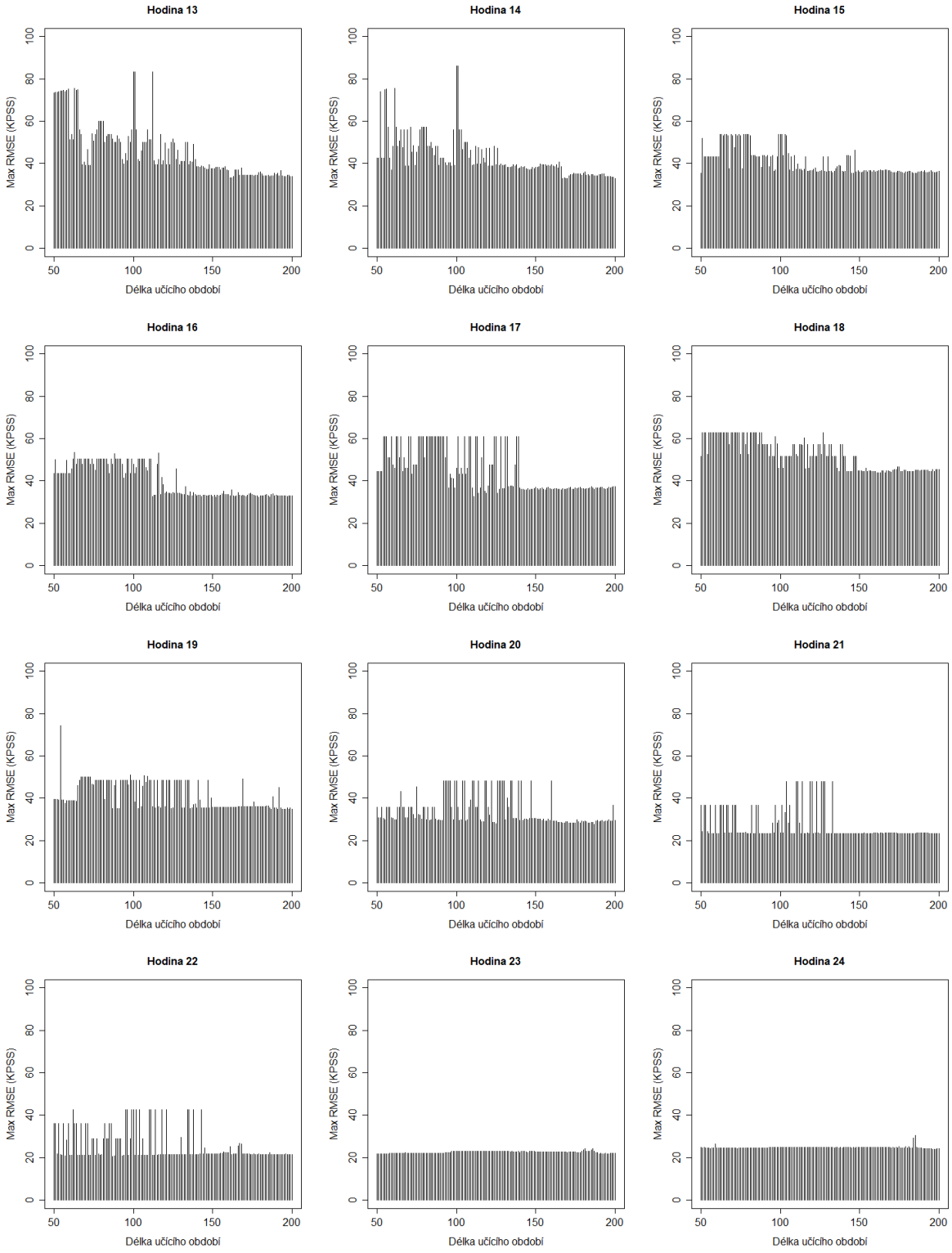
Obr. 3: Průměrná $RMSE$ v závislosti na délce učícího období pro $KPSS$–test.


Hodina 1
Hodina 2
Hodina 3
Hodina 4
Hodina 5
Hodina 6
Hodina 7
Hodina 8
Hodina 9
Hodina 10
Hodina 11
Hodina 12
Max RMSE (ADF)
Délka učícího období
0
20
40
60
80
100
50
100
150
200

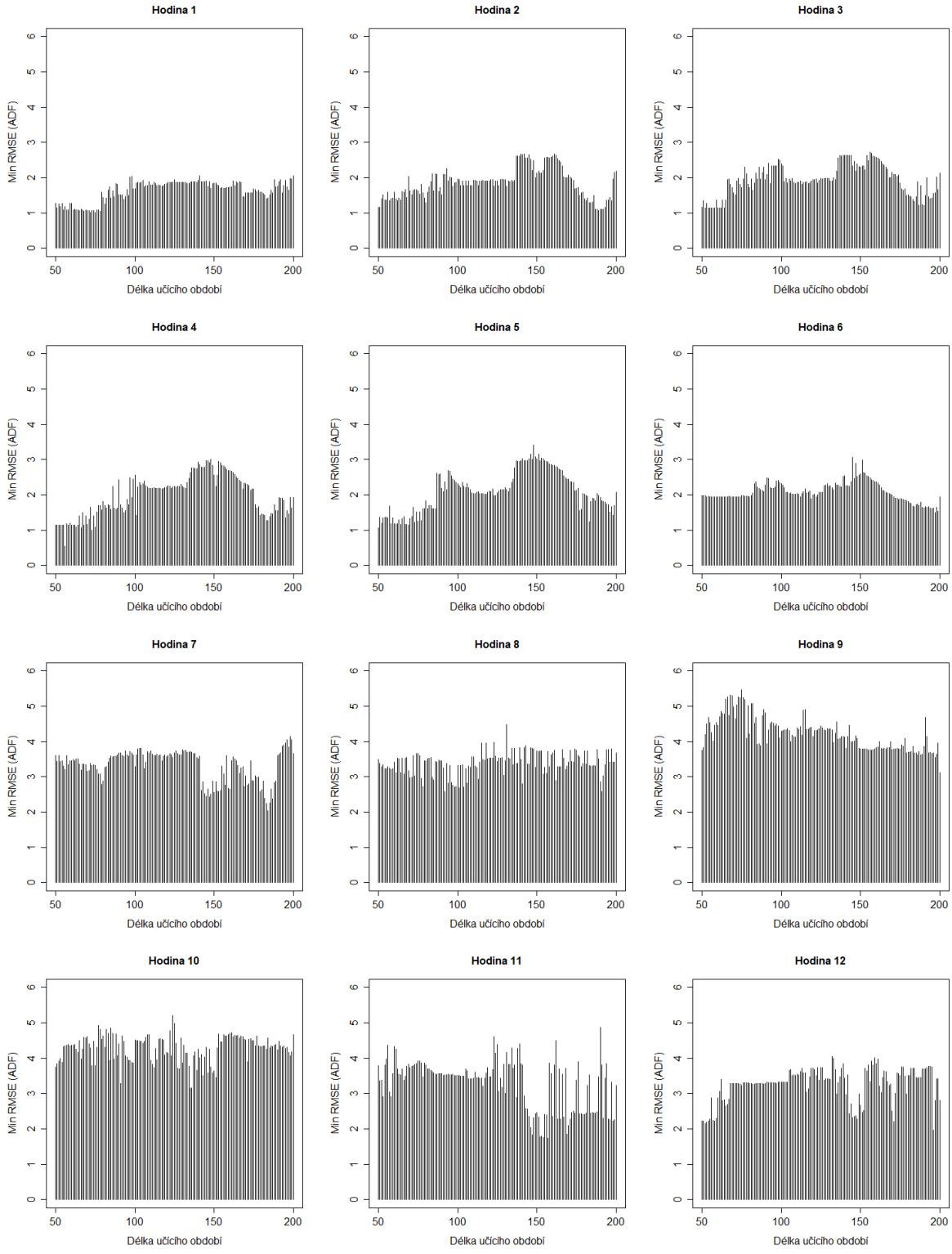


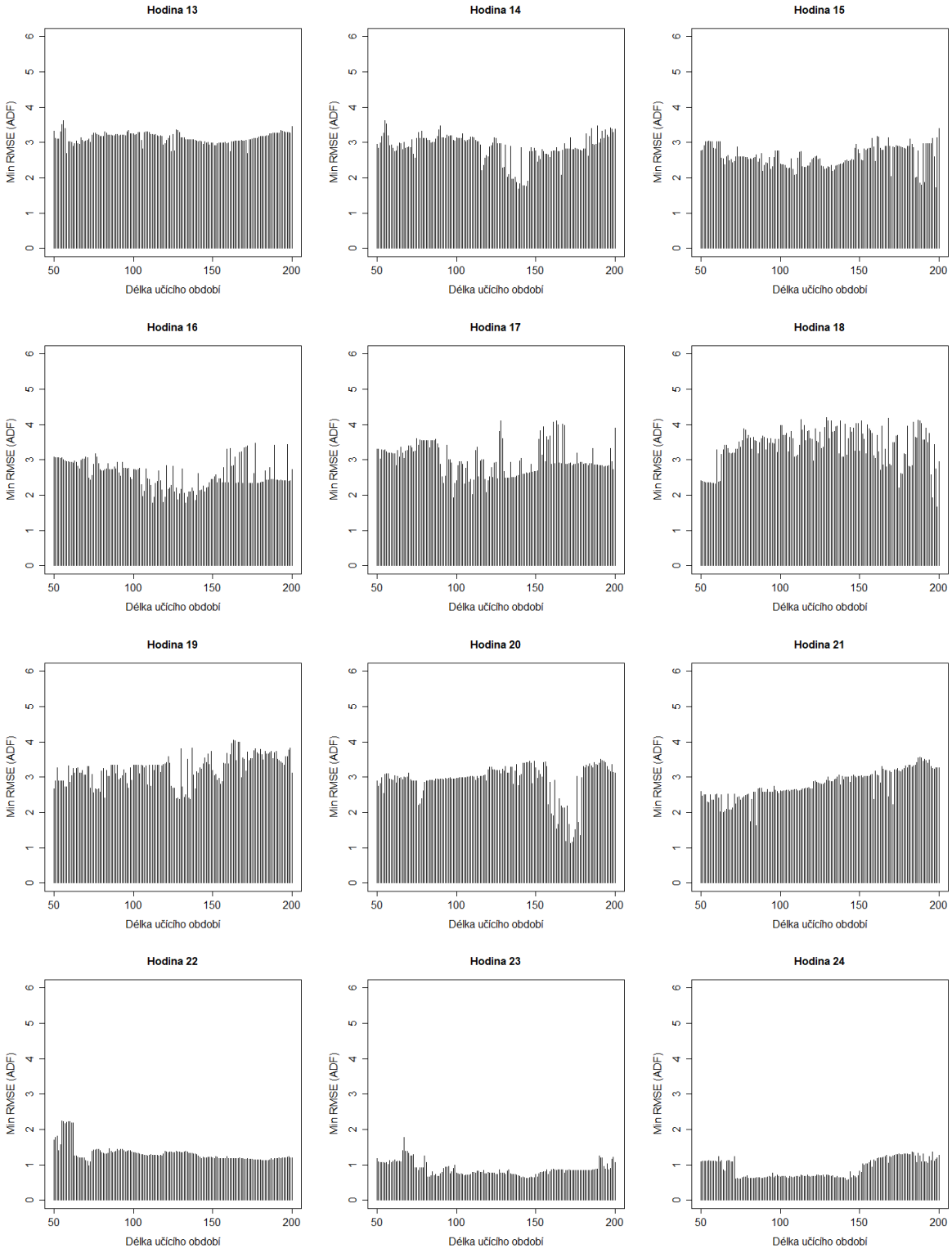
Obr. 4: Maximální $RMSE$ v závislosti na délce učícího období pro ADF–test.


Hodina 1
Hodina 2
Hodina 3
Hodina 4
Hodina 5
Hodina 6
Hodina 7
Hodina 8
Hodina 9
Hodina 10
Hodina 11
Hodina 12
Max RMSE (KPSS)
Délka učícího období
0
20
40
60
80
100
50
100
150
200

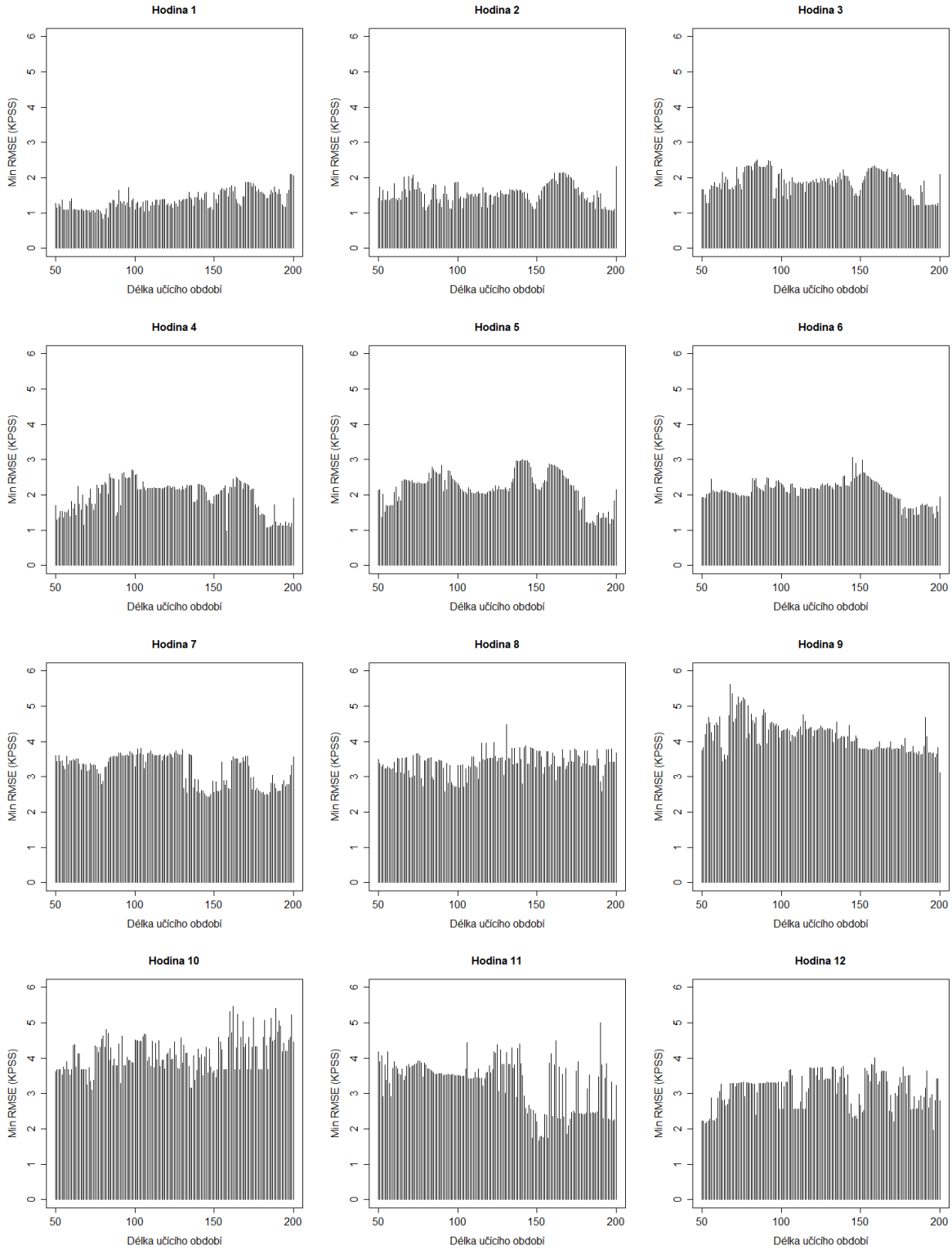


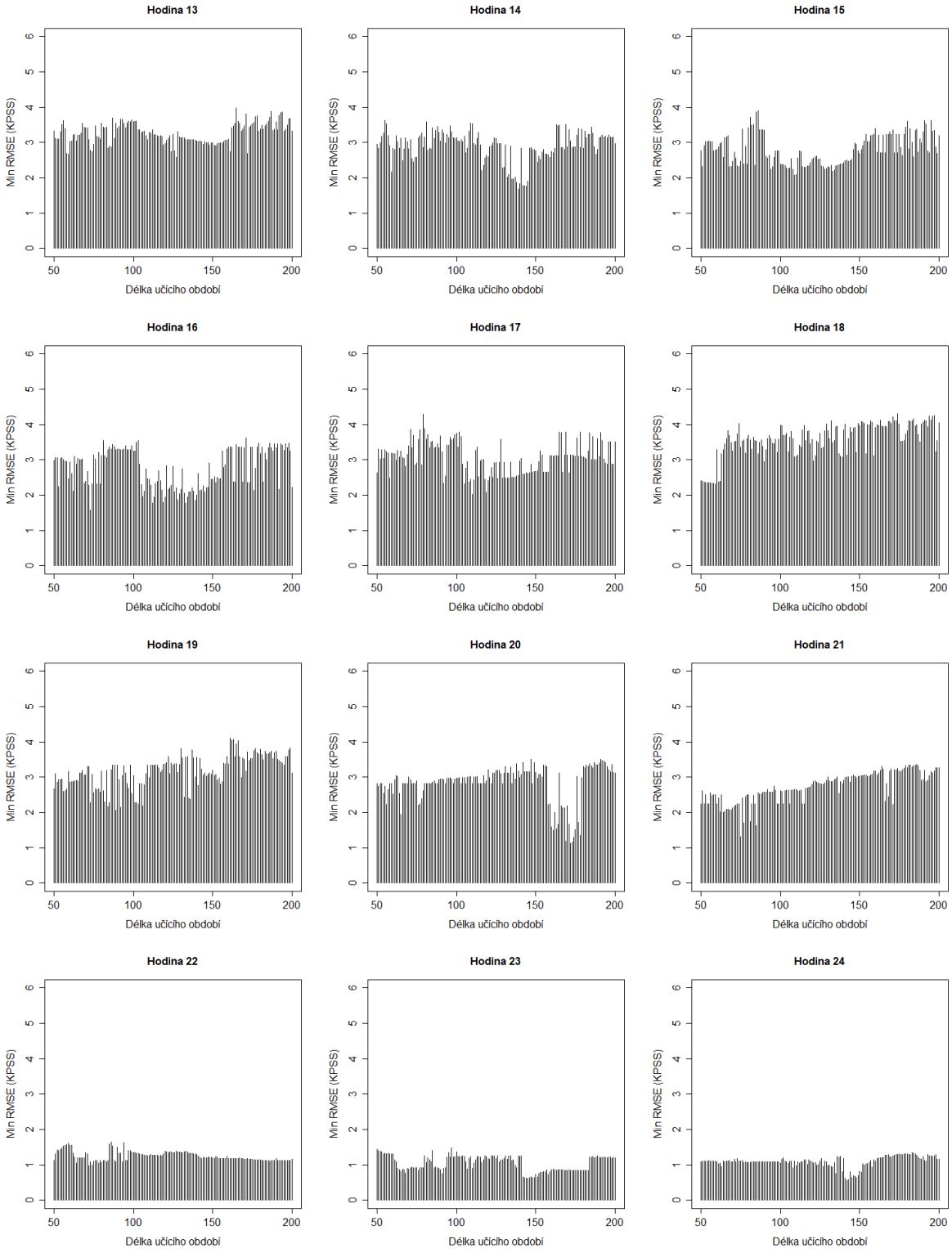
Obr. 5: Maximální $RMSE$ v závislosti na délce učícího období pro $KPSS$–test.


Hodina 1
Hodina 2
Hodina 3
Hodina 4
Hodina 5
Hodina 6
Hodina 7
Hodina 8
Hodina 9
Hodina 10
Hodina 11
Hodina 12
Min RMSE (ADF)
Délka učícího období
0
1
2
3
4
5
6
50
100
150
200

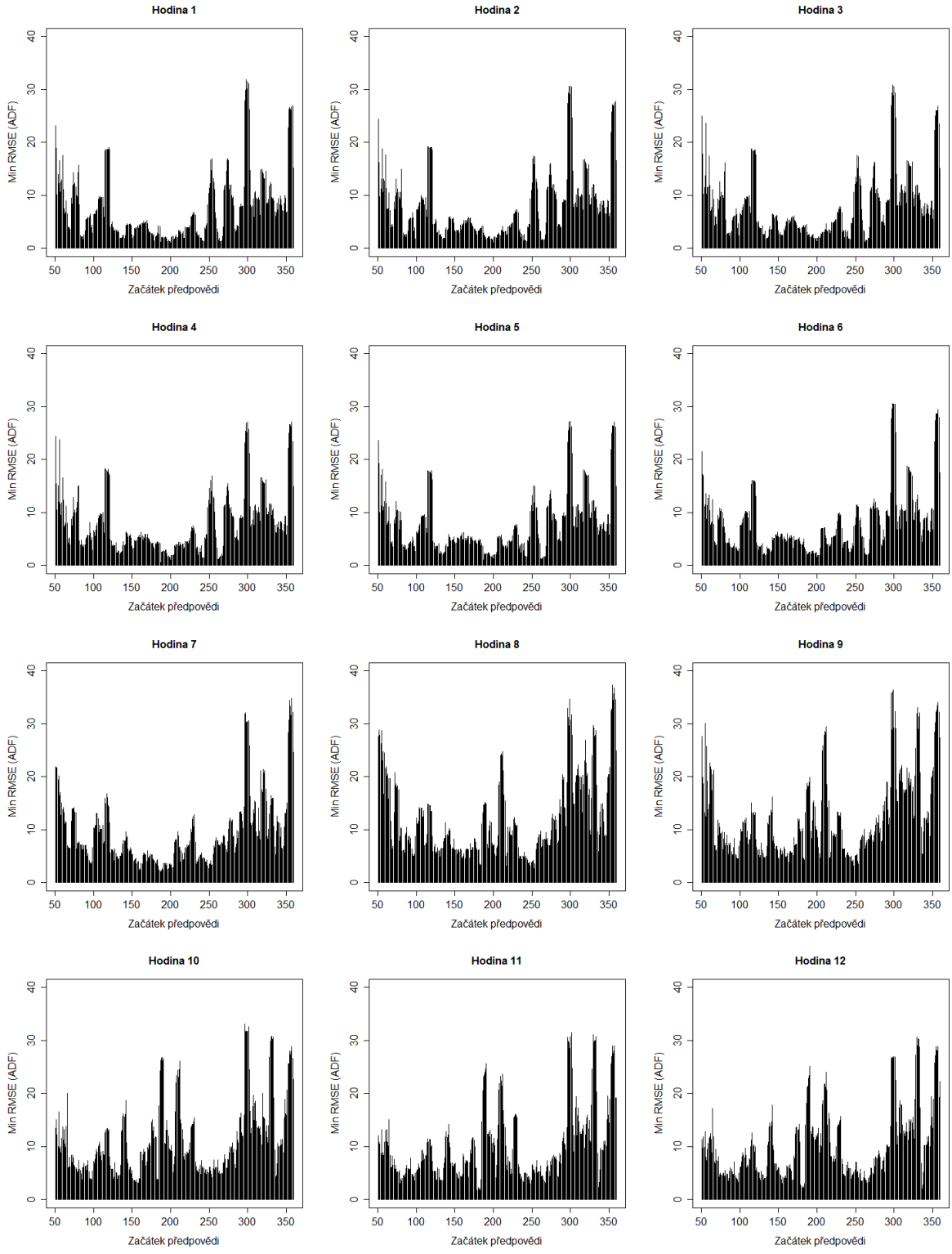


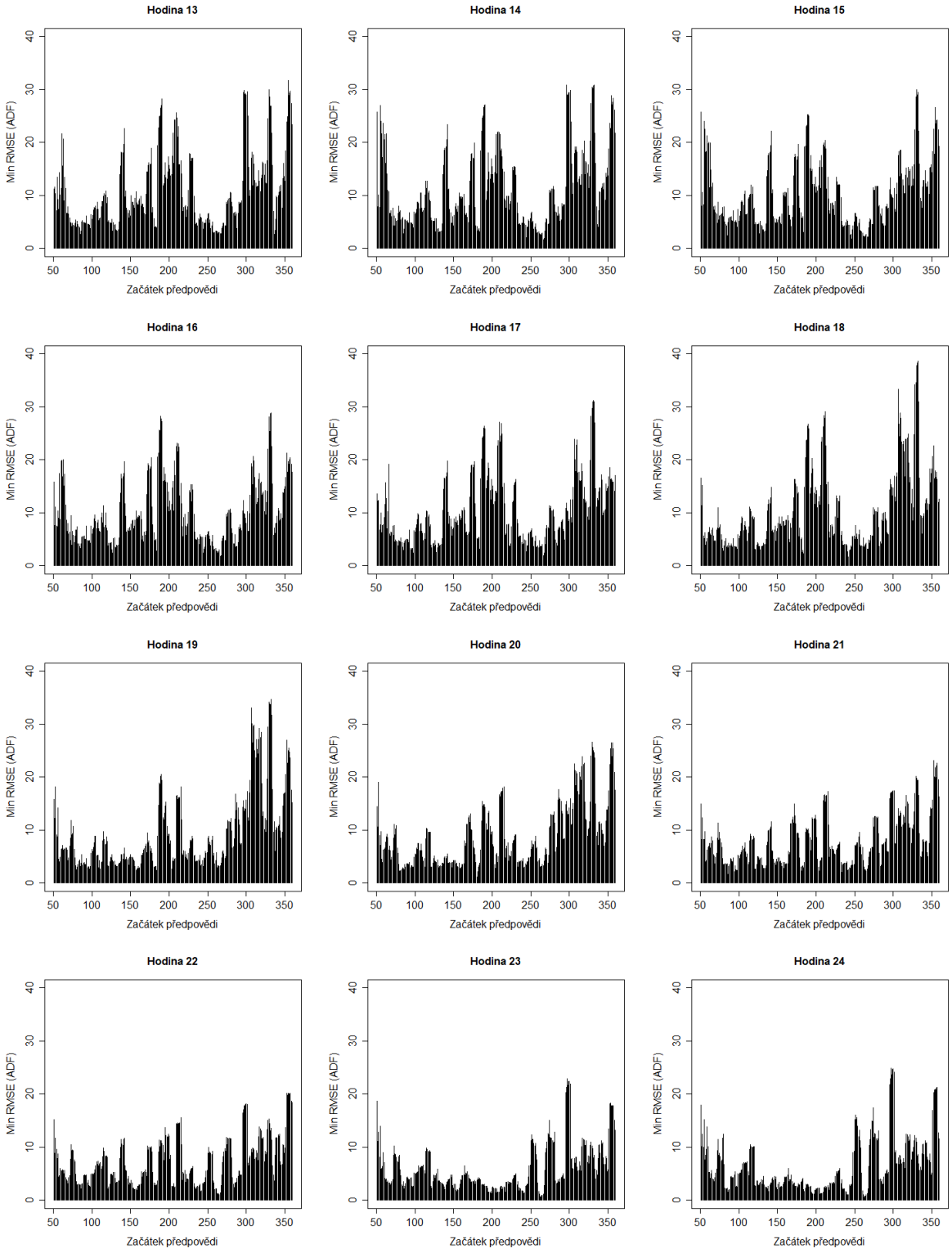
Obr. 6: Minimální $RMSE$ v závislosti na délce učícího období pro ADF–test.


Hodina 1
Hodina 2
Hodina 3
Hodina 4
Hodina 5
Hodina 6
Hodina 7
Hodina 8
Hodina 9
Hodina 10
Hodina 11
Hodina 12
Min RMSE (KPSS)
Délka učícího období
0
1
2
3
4
5
6
50
100
150
200

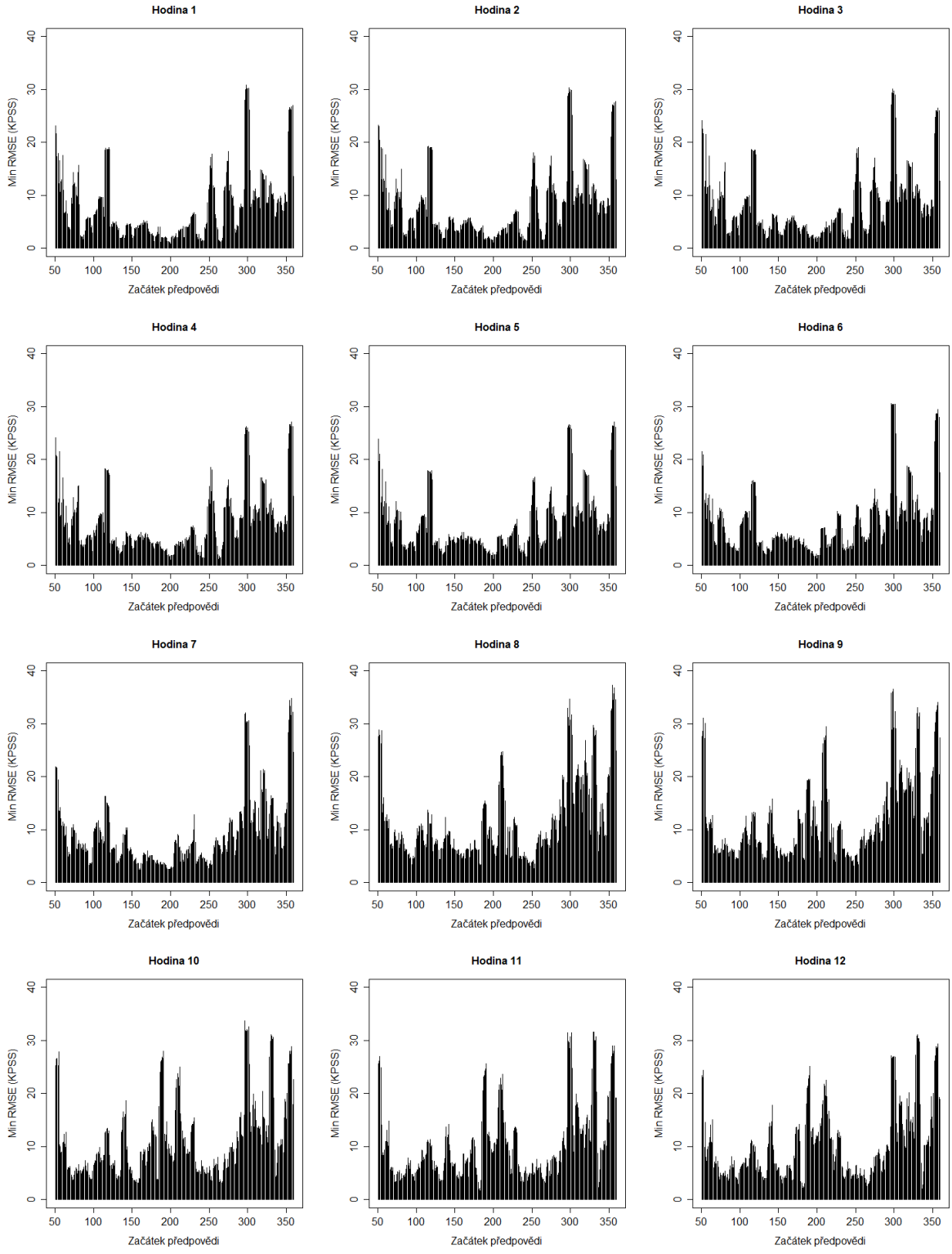


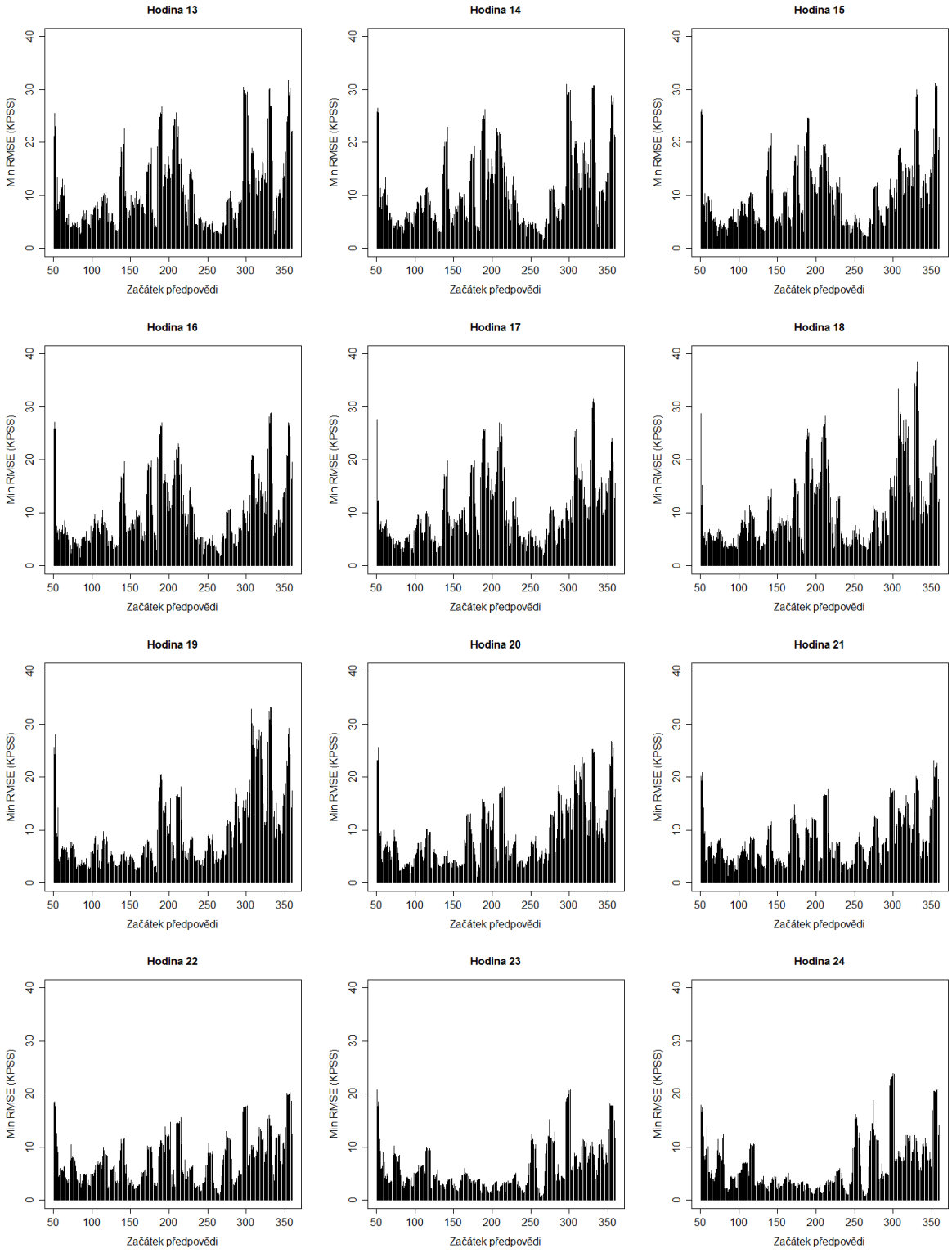
Obr. 7: Minimální $RMSE$ v závislosti na délce učícího období pro $KPSS$–test.


Hodina 1
Hodina 2
Hodina 3
Hodina 4
Hodina 5
Hodina 6
Hodina 7
Hodina 8
Hodina 9
Hodina 10
Hodina 11
Hodina 12
Min RMSE (ADF)
Začátek předpovědi
0
10
20
30
40
50
100
150
200
250
300
350

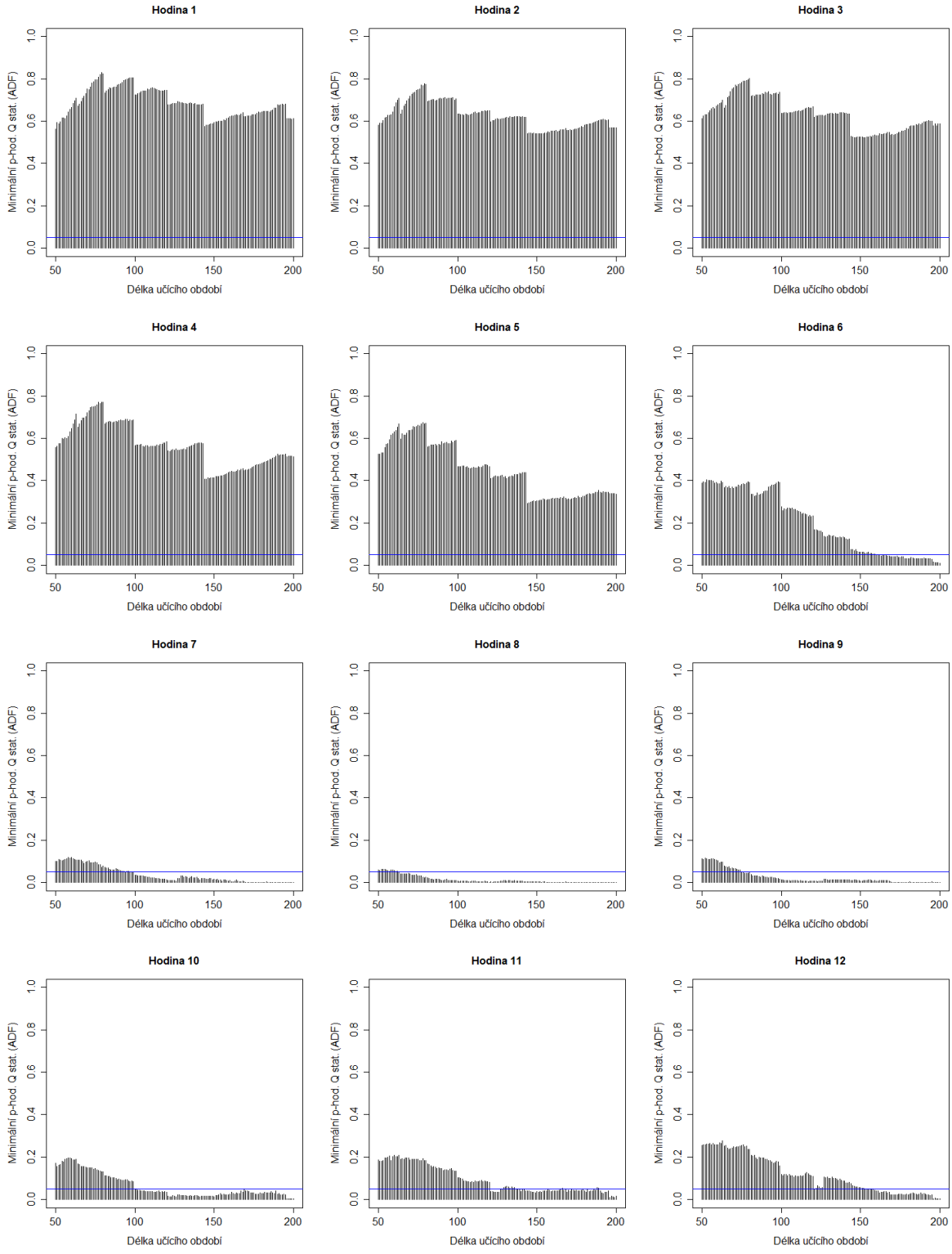


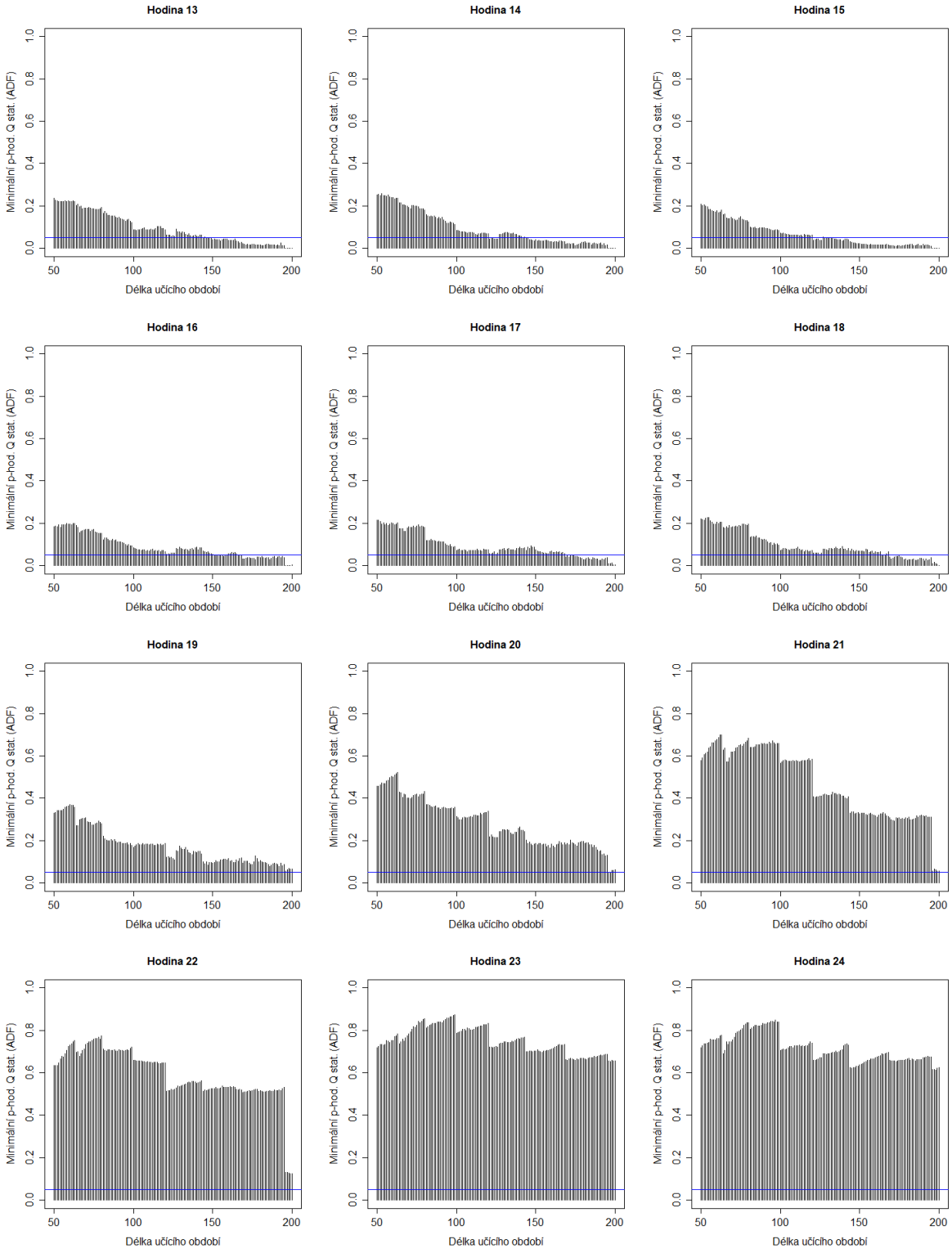
Obr. 8: Minimální $RMSE$ v závislosti na počátečním dni předpovědi pro ADF–test.


Hodina 1
Hodina 2
Hodina 3
Hodina 4
Hodina 5
Hodina 6
Hodina 7
Hodina 8
Hodina 9
Hodina 10
Hodina 11
Hodina 12
Min RMSE (KPSS)
Začátek předpovědi
0
10
20
30
40
50
100
150
200
250
300
350

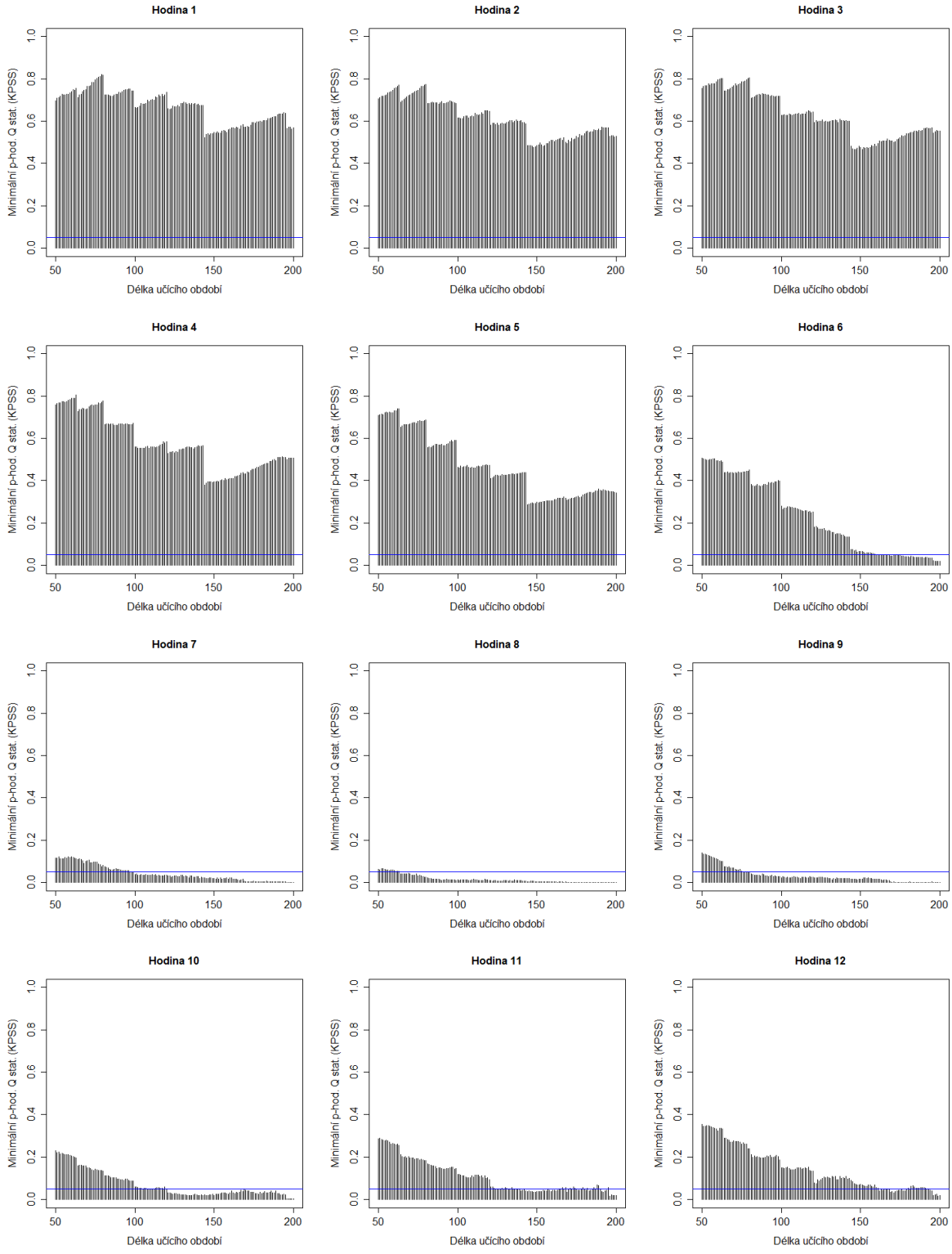


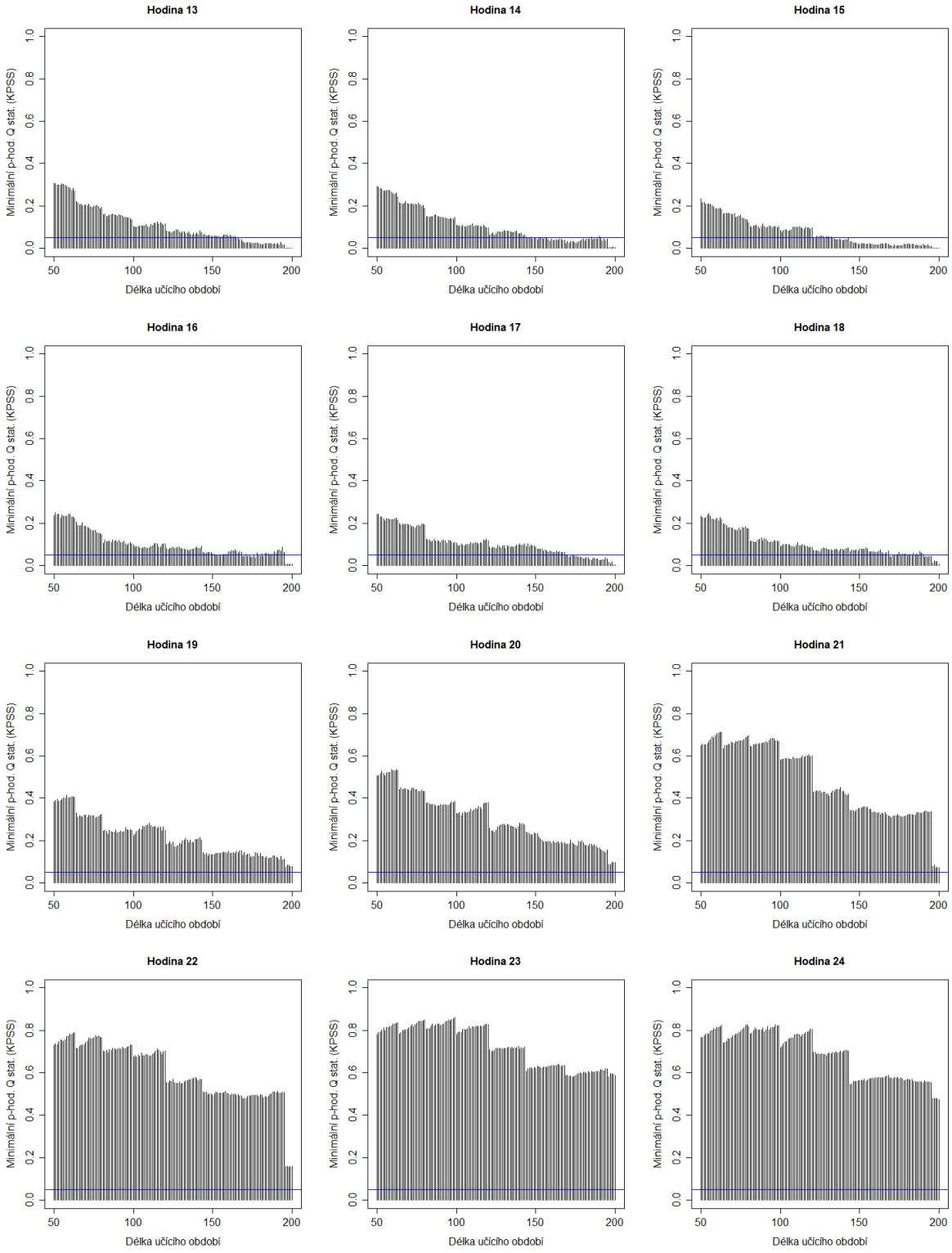
Obr. 9: Minimální $RMSE$ v závislosti na počátečním dni předpovědi pro $KPSS$–test.


Hodina 1
Hodina 2
Hodina 3
Hodina 4
Hodina 5
Hodina 6
Hodina 7
Hodina 8
Hodina 9
Hodina 10
Hodina 11
Hodina 12
Minimální p-hod. Q stat. (ADF)
Délka učícího období
0.0
0.2
0.4
0.6
0.8
1.0
50
100
150
200

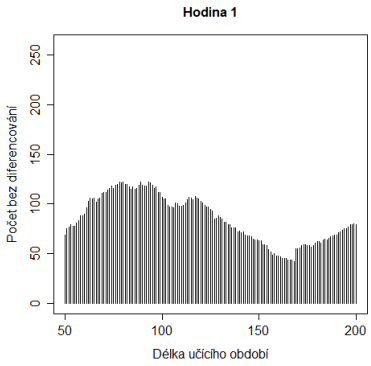


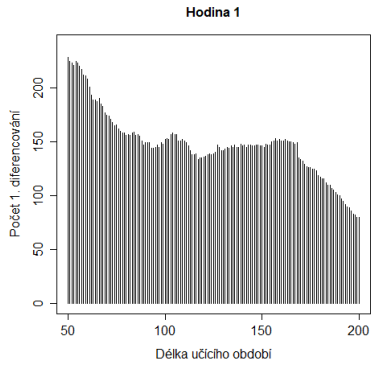
Obr. 10: Minimální p–hodnota Q–testu reziduí předpovědního modelu v závislosti na délce učícího období pro ADF–test.

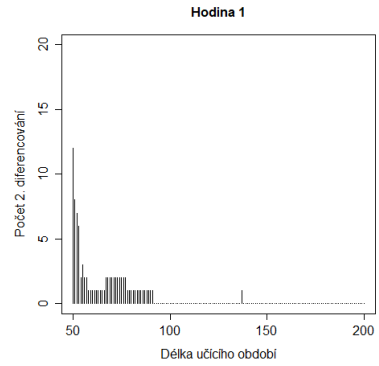

Hodina 1
Hodina 2
Hodina 3
Hodina 4
Hodina 5
Hodina 6
Hodina 7
Hodina 8
Hodina 9
Hodina 10
Hodina 11
Hodina 12
Minimální p-hod. Q stat. (KPSS)
Délka učícího období
0.0
0.2
0.4
0.6
0.8
1.0
50
100
150
200

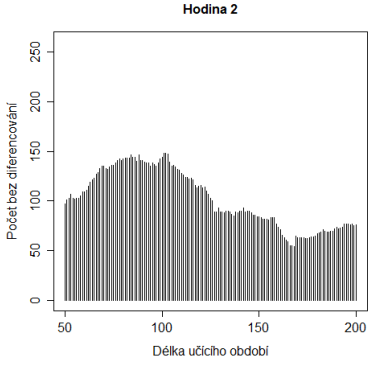


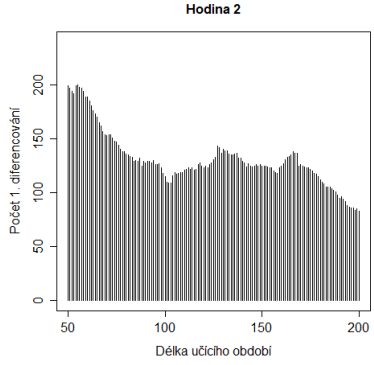
Obr. 11: Minimální p–hodnota Q–testu reziduí předpovědního modelu v závislosti na délce učícího období pro $KPSS$–test.

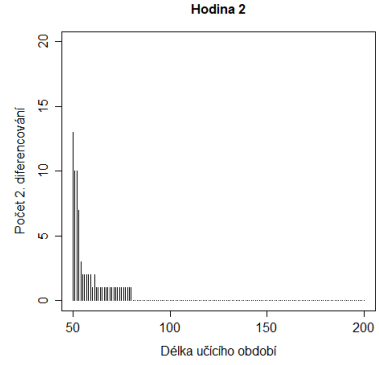

Hodina 1
Počet bez diferencování
Délka učícího období

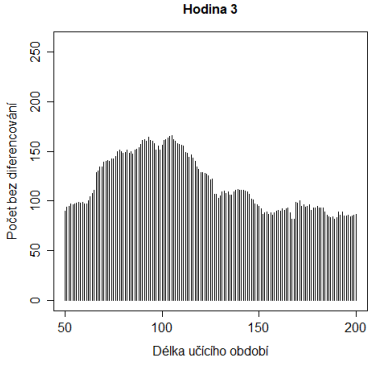

Hodina 1
Počet 1. diferencování
Délka učícího období

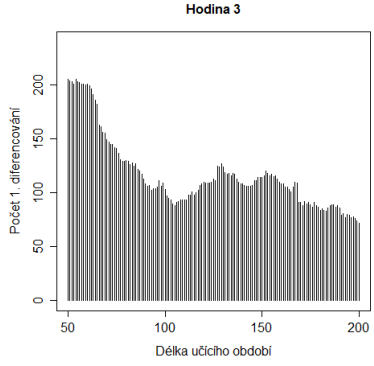

Hodina 1
Počet 2. diferencování
Délka učícího období

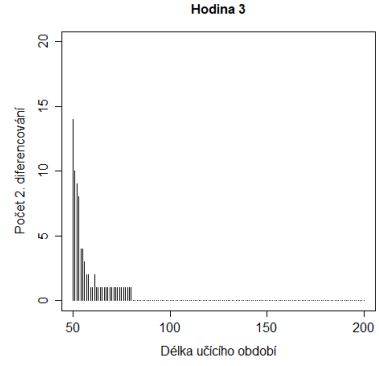

Hodina 2
Počet bez diferencování
Délka učícího období

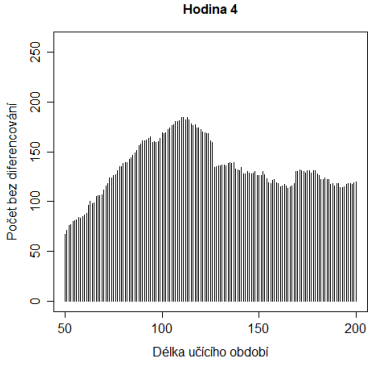

Hodina 2
Počet 1. diferencování
Délka učícího období

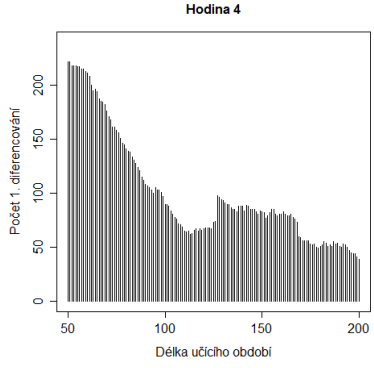

Hodina 2
Počet 2. diferencování
Délka učícího období

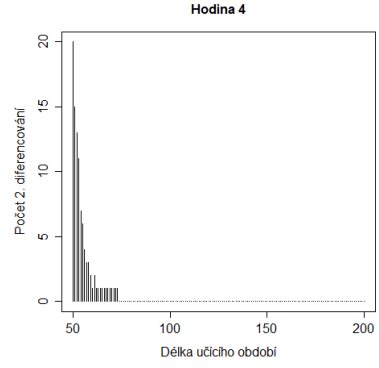

Hodina 3
Počet bez diferencování
Délka učícího období

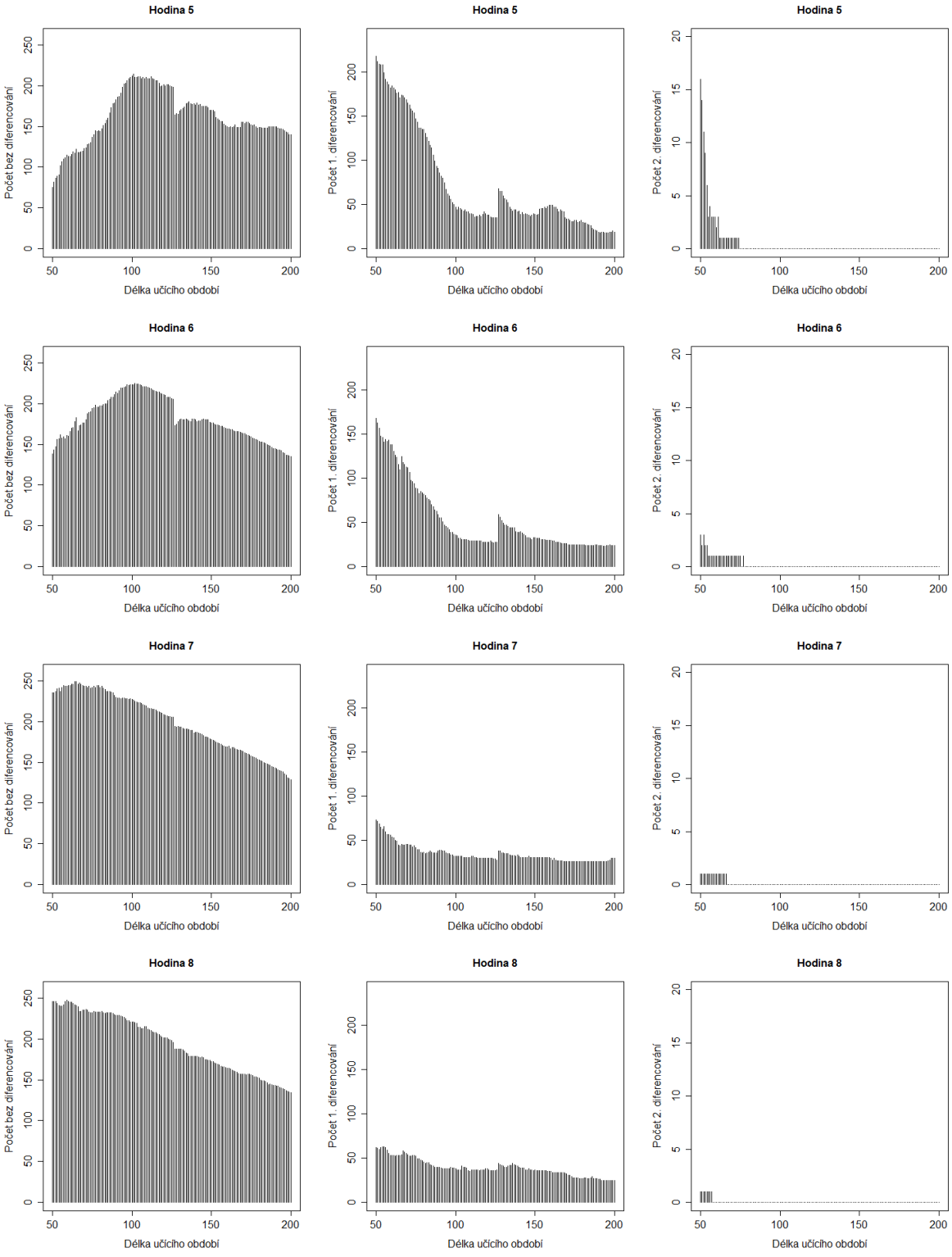

Hodina 3
Počet 1. diferencování
Délka učícího období

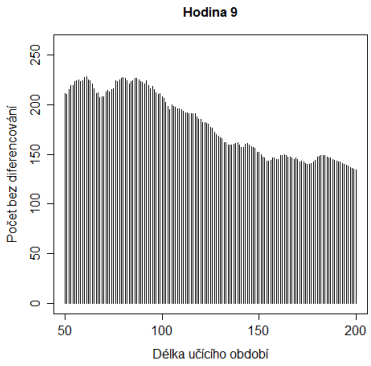

Hodina 3
Počet 2. diferencování
Délka učícího období

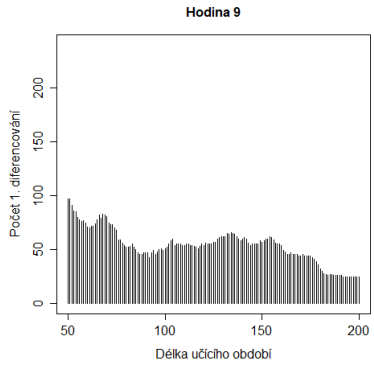

Hodina 4
Počet bez diferencování
Délka učícího období

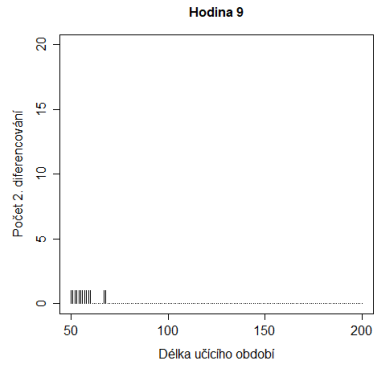

Hodina 4
Počet 1. diferencování
Délka učícího období

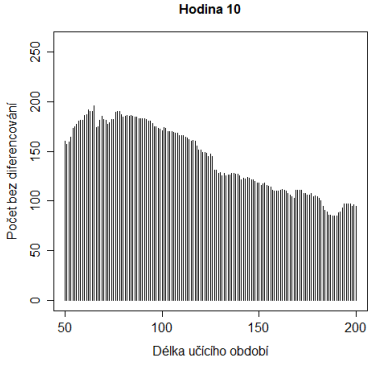

Hodina 4
Počet 2. diferencování
Délka učícího období

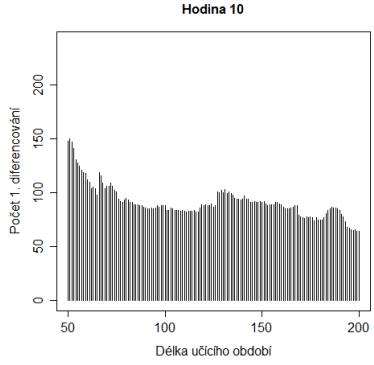

Hodina 5
Počet bez diferencování
Délka učícího období
Hodina 5
Počet 1. diferencování
Délka učícího období
Hodina 5
Počet 2. diferencování
Délka učícího období
Hodina 6
Počet bez diferencování
Délka učícího období
Hodina 6
Počet 1. diferencování
Délka učícího období
Hodina 6
Počet 2. diferencování
Délka učícího období
Hodina 7
Počet bez diferencování
Délka učícího období
Hodina 7
Počet 1. diferencování
Délka učícího období
Hodina 7
Počet 2. diferencování
Délka učícího období
Hodina 8
Počet bez diferencování
Délka učícího období
Hodina 8
Počet 1. diferencování
Délka učícího období
Hodina 8
Počet 2. diferencování
Délka učícího období

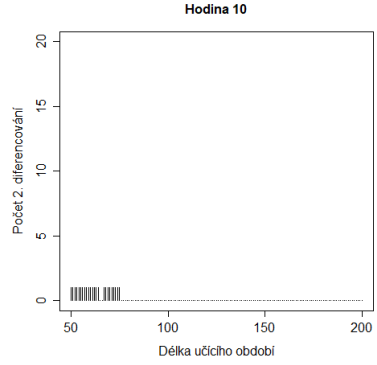

Hodina 9
Počet bez diferencování
Délka učícího období

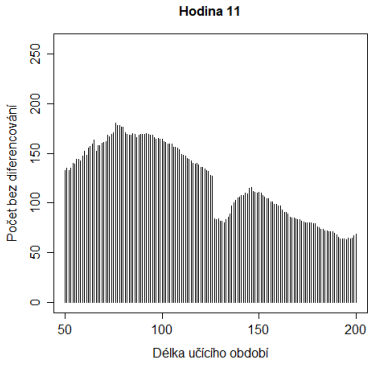

Hodina 9
Počet 1. diferencování
Délka učícího období

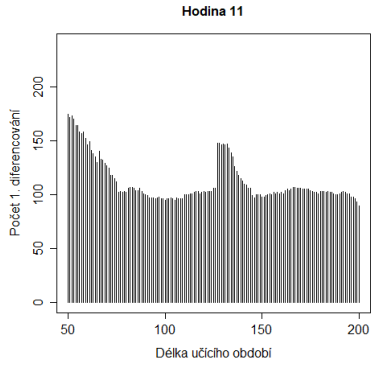

Hodina 9
Počet 2. diferencování
Délka učícího období

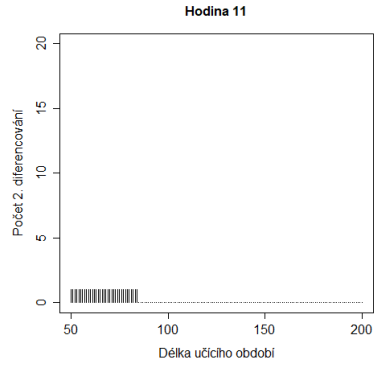

Hodina 10
Počet bez diferencování
Délka učícího období

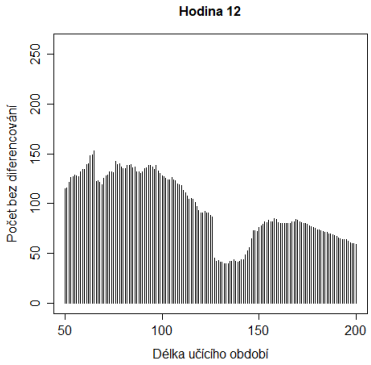

Hodina 10
Počet 1. diferencování
Délka učícího období

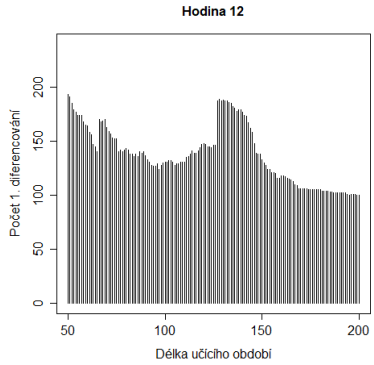

Hodina 10
Počet 2. diferencování
Délka učícího období

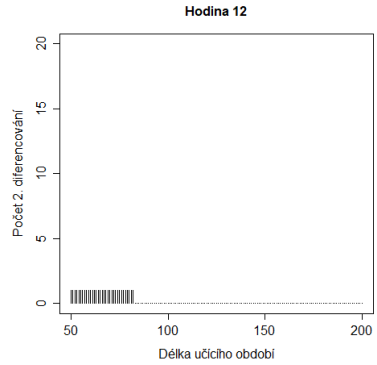

Hodina 11
Počet bez diferencování
Délka učícího období

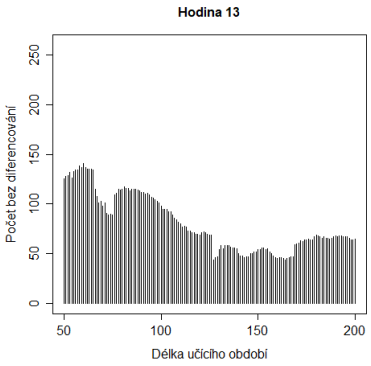

Hodina 11
Počet 1. diferencování
Délka učícího období

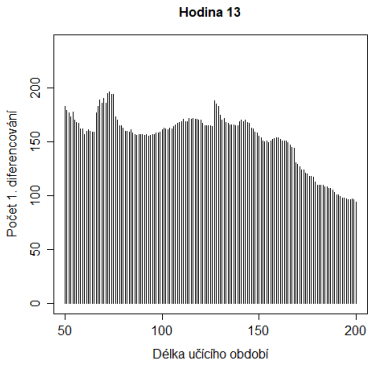

Hodina 11
Počet 2. diferencování
Délka učícího období

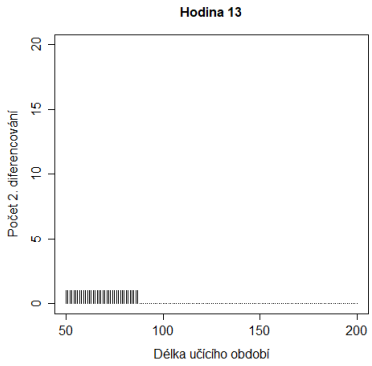

Hodina 12
Počet bez diferencování
Délka učícího období

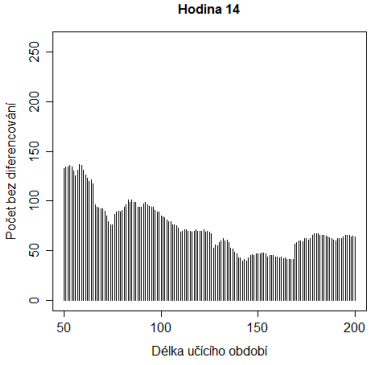

Hodina 12
Počet 1. diferencování
Délka učícího období

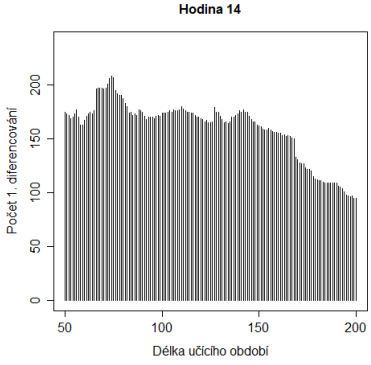

Hodina 12
Počet 2. diferencování
Délka učícího období

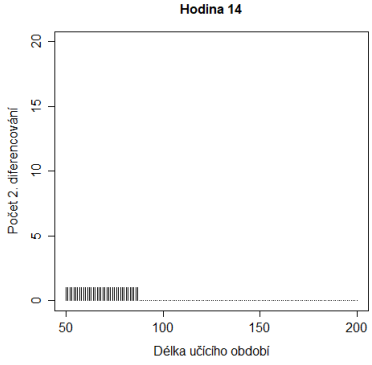

Hodina 13
Počet bez diferencování
Délka učícího období

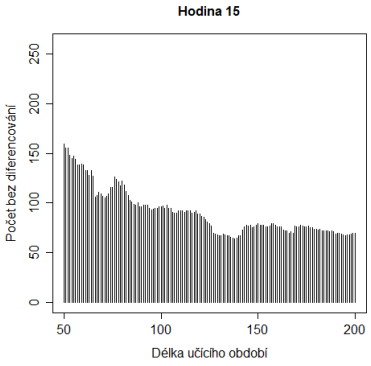

Hodina 13
Počet 1. diferencování
Délka učícího období

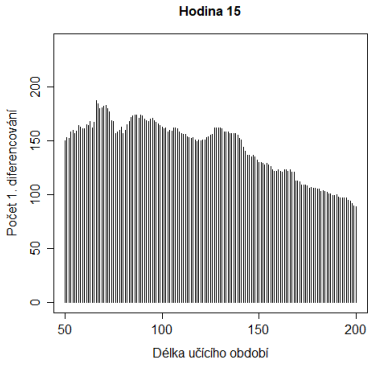

Hodina 13
Počet 2. diferencování
Délka učícího období

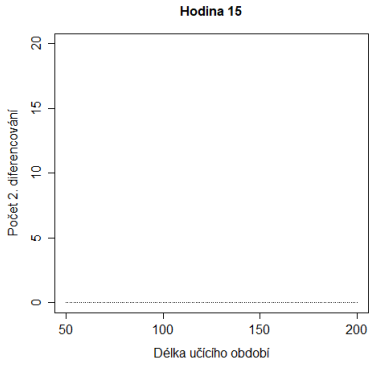

Hodina 14
Počet bez diferencování
Délka učícího období

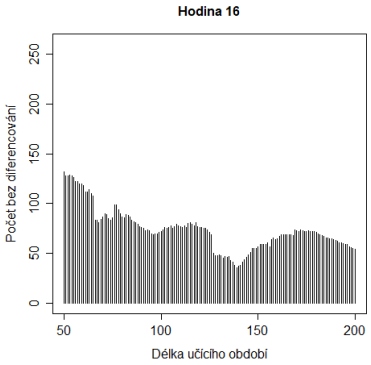

Hodina 14
Počet 1. diferencování
Délka učícího období

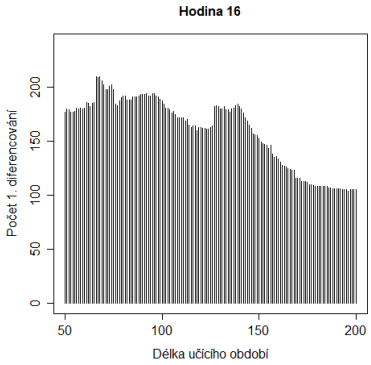

Hodina 14
Počet 2. diferencování
Délka učícího období

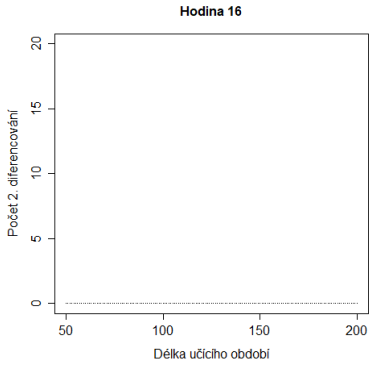

Hodina 15
Počet bez diferencování
Délka učícího období

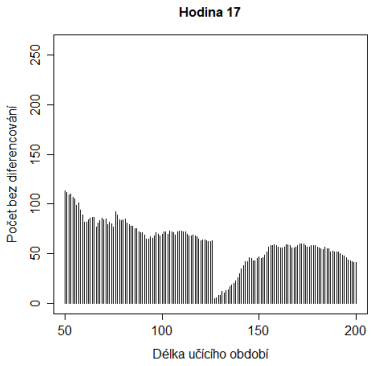

Hodina 15
Počet 1. diferencování
Délka učícího období

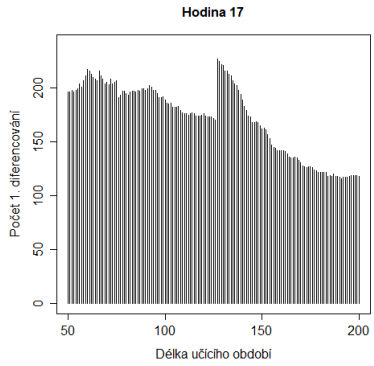

Hodina 15
Počet 2. diferencování
Délka učícího období

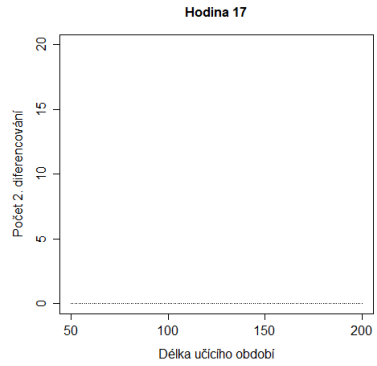

Hodina 16
Počet bez diferencování
Délka učícího období

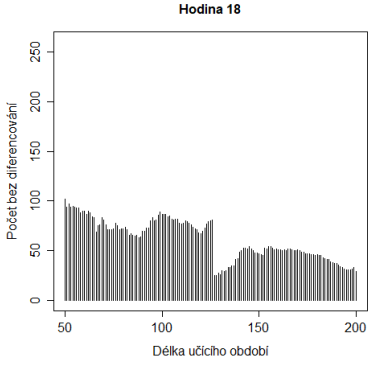

Hodina 16
Počet 1. diferencování
Délka učícího období

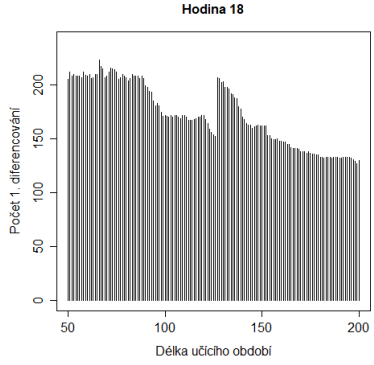

Hodina 16
Počet 2. diferencování
Délka učícího období

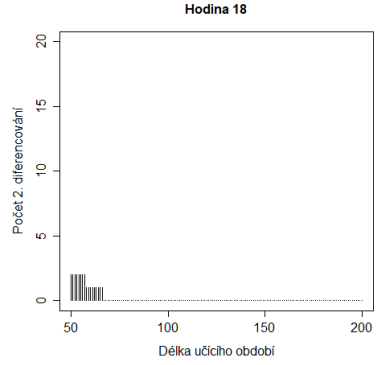

Hodina 17
Počet bez diferencování
Délka učícího období

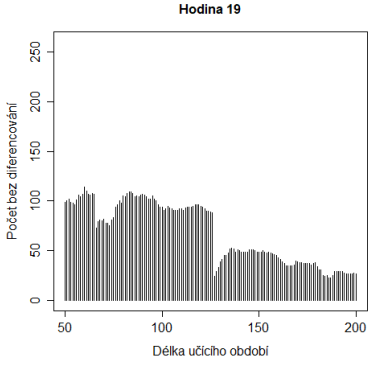

Hodina 17
Počet 1. diferencování
Délka učícího období

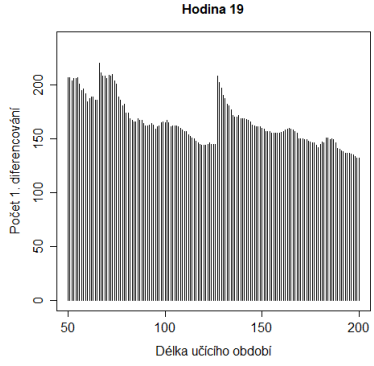

Hodina 17
Počet 2. diferencování
Délka učícího období

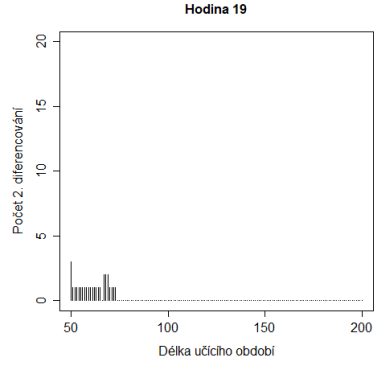

Hodina 18
Počet bez diferencování
Délka učícího období

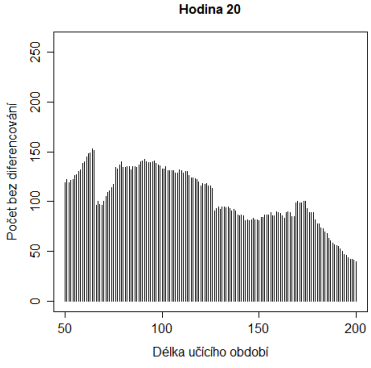

Hodina 18
Počet 1. diferencování
Délka učícího období

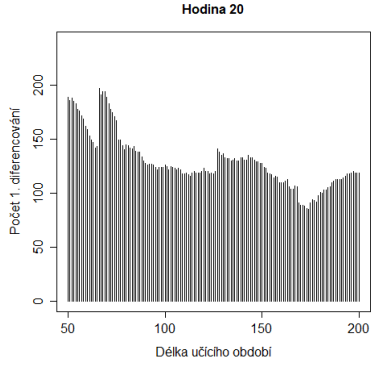

Hodina 18
Počet 2. diferencování
Délka učícího období

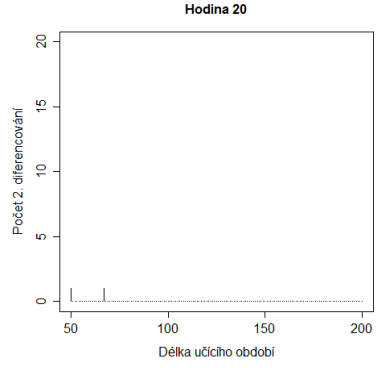

Hodina 19
Počet bez diferencování
Délka učícího období

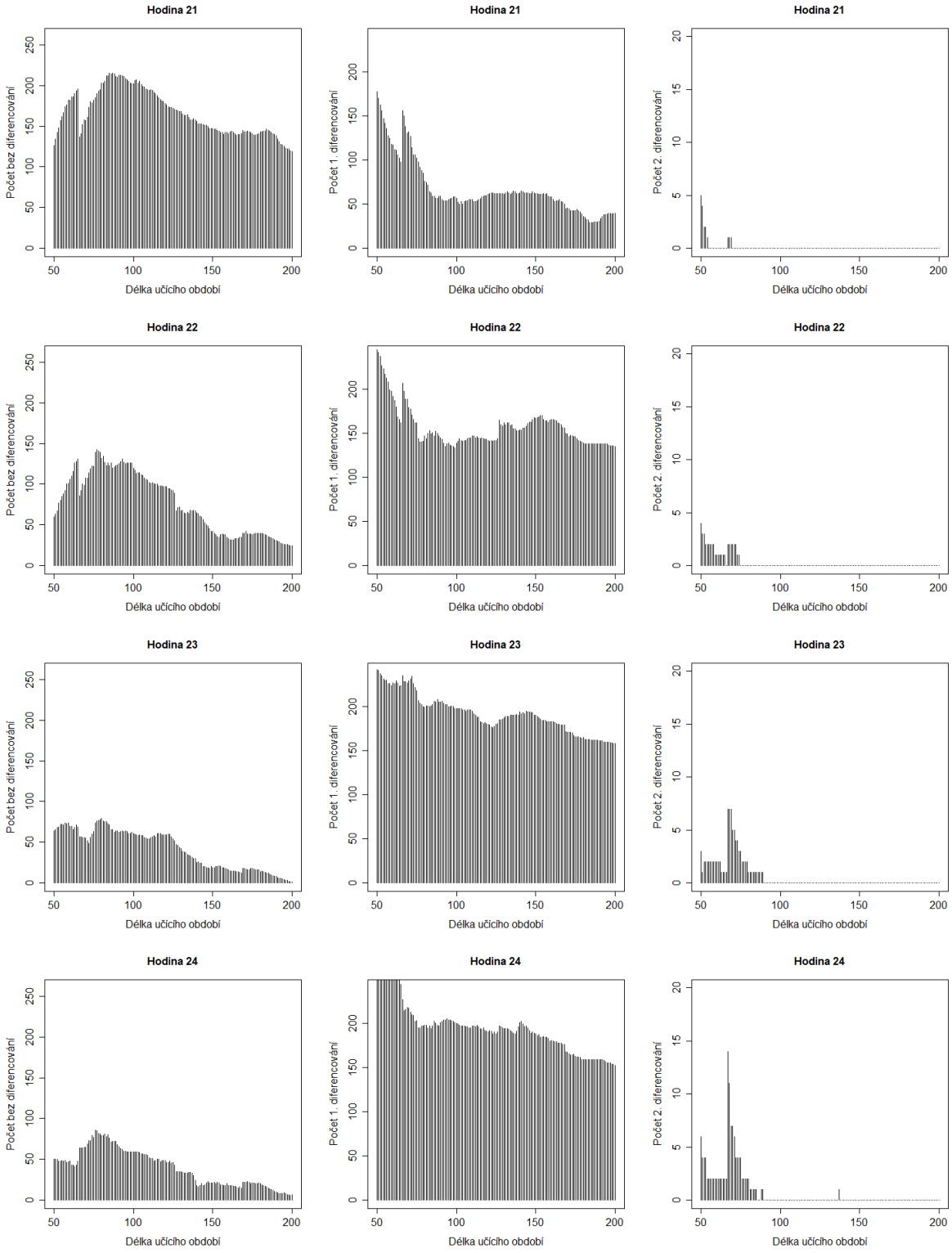

Hodina 19
Počet 1. diferencování
Délka učícího období


Hodina 19
Počet 2. diferencování
Délka učícího období


Hodina 20
Počet bez diferencování
Délka učícího období


Hodina 20
Počet 1. diferencování
Délka učícího období


Hodina 20
Počet 2. diferencování
Délka učícího období



Obr. 12: Počty diferencování v závislosti na délce učícího období. Řady učících období byly diferencovány vždy, když jeden z testů $KPSS$ nebo ADF detekoval nestacionaritu.

Tabulka 1: Predikce a skutečné hodnoty třídenní předpovědi MCP pro 265. až 267. den (případ $ARIMA$ s nejmenší chybou).

Hodina	1	2	3	4	5	6	7	8	9	10	11	12
Predikce	30,89	29,11	27,39	26,35	26,42	29,05	35,68	49,27	45,27	44,72	43,13	41,34
Skutečnost	31,44	30,19	28,49	27,7	28,95	31	40,33	48,3	49,82	48,08	44,88	43

Hodina	13	14	15	16	17	18	19	20	21	22	23	24
Predikce	38,50	34,95	35,23	39,13	43,12	43,56	45,09	45,14	46,50	38,87	35,90	31,84
Skutečnost	38,42	37,1	35	34,8	36,4	42,33	46,5	49,64	48,3	40,8	37,5	33,68

Hodina	25	26	27	28	29	30	31	32	33	34	35	36
Predikce	29,99	28,18	26,50	25,50	25,50	27,60	31,95	42,20	43,00	42,43	40,48	37,44
Skutečnost	34,3	32,25	31,42	30,8	30,87	31,5	33,6	36,4	40,5	41,8	40,7	37,3

Hodina	37	38	39	40	41	42	43	44	45	46	47	48
Predikce	37,91	33,80	34,44	36,14	39,07	38,55	43,30	41,95	44,81	38,87	35,13	31,41
Skutečnost	35,1	33,5	32,8	33,56	34,79	38,5	43,2	46,38	44,1	39	36	32

Hodina	49	50	51	52	53	54	55	56	57	58	59	60
Predikce	29,54	27,71	26,04	25,04	25,02	26,90	31,95	36,46	43,00	42,43	40,48	36,84
Skutečnost	30,25	27,48	27,34	26,81	26,82	26,83	26,6	29	32,7	34,27	35	35,8

Hodina	61	62	63	64	65	66	67	68	69	70	71	72
Predikce	37,56	33,74	34,44	31,68	37,71	38,73	42,46	41,95	44,14	38,87	34,89	30,62
Skutečnost	34,39	32,37	31,4	31	31	33,5	40,25	44,08	43,8	38,93	36,1	31,5

Tabulka 2: Predikce a skutečné hodnoty třídenní předpovědi MCP pro 300. až 302. den (případ $ARIMA$ s největší chybou).

Hodina	1	2	3	4	5	6	7	8	9	10	11	12
Predikce	30,28	30,61	27,05	27,69	27,50	27,89	37,72	41,50	49,46	44,08	41,37	39,02
Skutečnost	24,5	22,6	17,7	17,56	18,5	23,09	33,2	39,5	40,88	38,9	36,5	34,5

Hodina	13	14	15	16	17	18	19	20	21	22	23	24
Predikce	35,40	34,05	34,34	37,23	36,87	45,43	50,61	50,98	43,03	37,24	34,03	29,17
Skutečnost	29,5	25,11	23,3	26	32,72	38,69	44,8	45,6	39,51	35,5	31,7	26,5

Hodina	25	26	27	28	29	30	31	32	33	34	35	36
Predikce	25,72	25,41	25,75	25,44	25,79	26,54	33,87	44,02	46,16	41,67	39,39	37,73
Skutečnost	5	1	0,99	2	0,7	0,1	4,7	5,72	9,67	15,2	14	13

Hodina	37	38	39	40	41	42	43	44	45	46	47	48
Predikce	34,46	32,62	34,34	39,12	35,05	45,03	51,34	50,98	42,73	37,80	33,40	28,59
Skutečnost	9,81	10,01	10	10	10	4,34	13,63	14,2	2,01	-6	-20	-30

Hodina	49	50	51	52	53	54	55	56	57	58	59	60
Predikce	28,55	25,05	25,09	24,59	24,94	25,96	33,87	42,97	46,16	41,67	39,39	37,73
Skutečnost	-50	-50	-50	-40	-40	-50	-50	-50	-50	-40	-40	-32,01

Hodina	61	62	63	64	65	66	67	68	69	70	71	72
Predikce	34,46	32,62	34,34	31,46	35,05	45,03	51,34	50,98	42,64	37,77	33,14	28,45
Skutečnost	-45,4	-50	9,99	17,99	26,58	21,09	29,79	33,2	22,1	20	13	4

Tabulka 3: Predikce a skutečné hodnoty jednodenní předpovědi MCP pro 300. den.

Hodina	1	2	3	4	5	6	7	8	9	10	11	12
Predikce	30,28	30,61	27,05	27,69	27,50	27,89	37,72	41,50	49,46	44,08	41,37	39,02
Skutečnost	24,5	22,6	17,7	17,56	18,5	23,09	33,2	39,5	40,88	38,9	36,5	34,5

Hodina	13	14	15	16	17	18	19	20	21	22	23	24
Predikce	35,40	34,05	34,34	37,23	36,87	45,43	50,61	50,98	43,03	37,24	34,03	29,17
Skutečnost	29,5	25,11	23,3	26	32,72	38,69	44,8	45,6	39,51	35,5	31,7	26,5

Tabulka 4: Predikce a skutečné hodnoty jednodenní předpovědi MCP pro 301. den.

Hodina	1	2	3	4	5	6	7	8	9	10	11	12
Predikce	24,83	21,94	22,00	20,91	21,21	24,33	32,12	44,22	41,67	40,01	37,86	36,49
Skutečnost	5	1	0,99	2	0,7	0,1	4,7	5,72	9,67	15,2	14	13

Hodina	13	14	15	16	17	18	19	20	21	22	23	24
Predikce	34,75	29,60	31,70	32,81	33,25	40,20	50,31	49,06	41,65	37,41	32,85	27,72
Skutečnost	9,81	10,01	10	10	10	4,34	13,63	14,2	2,01	-6	-20	-30

Tabulka 5: Predikce a skutečné hodnoty jednodenní předpovědi MCP pro 302. den.

Hodina	1	2	3	4	5	6	7	8	9	10	11	12
Predikce	18,90	16,53	14,91	12,70	11,45	14,61	21,83	28,37	27,92	32,43	30,53	30,03
Skutečnost	-50	-50	-50	-40	-40	-50	-50	-50	-50	-40	-40	-32,01

Hodina	13	14	15	16	17	18	19	20	21	22	23	24
Predikce	26,90	25,28	30,83	15,10	24,29	22,79	29,51	32,84	28,04	26,19	17,73	5,40
Skutečnost	-45,4	-50	9,99	17,99	26,58	21,09	29,79	33,2	22,1	20	13	4

Tabulka 6: Predikce a skutečné hodnoty třídenní předpovědi MCP $ARIMA$ a $ARIMA-G$ pro 325. až 327. den (největší zlepšení $ARIMA-G$).

Hodina	1	2	3	4	5	6	7	8	9	10	11	12
Predikce $ARIMA$	30,87	30,34	20,66	19,98	30,30	33,62	45,21	62,77	55,64	46,46	44,86	44,46
Predikce $ARIMA-G$	32,90	31,93	20,68	19,44	30,78	37,30	50,14	59,65	56,92	46,54	44,72	44,37
Skutečnost	34,18	33,89	33,2	33,5	34,2	38,01	49,1	61	64,5	63,64	64,51	65,68

Hodina	13	14	15	16	17	18	19	20	21	22	23	24
Predikce $ARIMA$	38,46	39,07	46,73	46,78	50,38	68,16	50,53	50,70	46,30	39,75	34,75	30,85
Predikce $ARIMA-G$	39,31	37,97	46,73	44,95	50,38	61,04	48,23	50,70	46,10	40,83	36,33	31,71
Skutečnost	63,8	63,29	60,88	60,8	62,66	67,39	60,99	51	40,6	35,3	32,8	28

Hodina	25	26	27	28	29	30	31	32	33	34	35	36
Predikce $ARIMA$	25,80	23,99	21,80	21,14	22,57	24,62	33,75	60,05	43,37	42,60	40,51	39,10
Predikce $ARIMA-G$	29,73	27,66	22,45	20,55	22,23	28,27	33,32	58,96	41,49	40,17	38,29	37,00
Skutečnost	24,4	20	16,27	14	17	22,96	30,91	40	39,5	36,22	35,5	34,9

Hodina	37	38	39	40	41	42	43	44	45	46	47	48
Predikce $ARIMA$	36,35	35,83	41,52	45,95	49,51	82,16	48,77	50,14	42,51	35,80	30,96	26,09
Predikce $ARIMA-G$	36,85	35,01	41,52	42,93	49,51	67,52	44,89	50,14	41,70	36,88	32,82	27,57
Skutečnost	32,82	33,68	34,63	38	44,2	51,39	47,98	43,7	35,09	31,4	29,3	22,91

Hodina	49	50	51	52	53	54	55	56	57	58	59	60
Predikce $ARIMA$	25,80	23,99	22,30	21,67	22,57	24,62	33,75	27,12	43,37	42,60	40,51	39,10
Predikce $ARIMA-G$	29,73	27,66	23,67	21,17	22,23	28,27	33,32	26,30	41,49	40,17	38,29	37,00
Skutečnost	23,9	17,33	11,5	11,75	14,7	23,9	29,61	36,68	35	32,92	24,7	23,9

Hodina	61	62	63	64	65	66	67	68	69	70	71	72
Predikce $ARIMA$	35,61	34,66	41,52	30,71	49,22	77,74	48,04	49,97	42,51	35,80	30,96	26,09
Predikce $ARIMA-G$	35,66	34,14	41,52	27,40	49,22	39,67	43,17	49,97	41,70	36,88	32,82	27,57
Skutečnost	23,9	23,9	23,9	29	34,43	38	36,57	35	35	30,1	25,5	20

Tabulka 7: Predikce a skutečné hodnoty třídenní předpovědi MCP $ARIMA$ a $ARIMA-G$ pro 312. až 314. den (největší zhoršení $ARIMA-G$).

Hodina	1	2	3	4	5	6	7	8	9	10	11	12
Predikce $ARIMA$	30,76	28,89	29,47	28,29	29,72	28,62	34,21	41,42	42,41	44,81	33,30	32,93
Predikce $ARIMA-G$	30,51	28,52	31,36	28,93	31,36	29,65	18,10	27,88	41,24	44,81	29,24	55,06
Skutečnost	35,2	34,25	33,91	33,95	34,9	36,36	50	68,39	76,55	72,04	70,95	73,26

Hodina	13	14	15	16	17	18	19	20	21	22	23	24
Predikce $ARIMA$	38,20	36,41	44,69	44,74	42,00	47,32	49,78	51,51	40,76	37,12	33,86	28,43
Predikce $ARIMA-G$	37,98	36,70	44,69	30,26	42,00	47,32	49,78	51,51	42,24	38,24	34,74	30,50
Skutečnost	71,41	70,39	71,2	80	88,6	110,6	114,52	90,8	66	52,7	48	40

Hodina	25	26	27	28	29	30	31	32	33	34	35	36
Predikce $ARIMA$	27,61	25,68	26,44	23,47	26,46	25,18	32,03	36,78	38,37	40,78	37,20	37,20
Predikce $ARIMA-G$	28,12	26,20	29,05	25,10	28,59	26,73	18,57	27,20	36,59	40,81	4,65	52,43
Skutečnost	38,5	37,5	37	36,5	38,62	41	53,53	72,6	73	66,91	63,7	60,5

Hodina	37	38	39	40	41	42	43	44	45	46	47	48
Predikce $ARIMA$	34,19	31,70	40,66	32,48	37,27	48,22	55,98	53,10	46,50	35,92	32,56	26,51
Predikce $ARIMA-G$	33,93	32,15	40,66	29,13	37,27	48,22	55,98	53,10	44,54	37,37	33,67	28,99
Skutečnost	56	56,08	55,53	56,57	57,79	65,8	67,35	57,01	45,54	38	34,83	30,67

Hodina	49	50	51	52	53	54	55	56	57	58	59	60
Predikce $ARIMA$	28,80	26,98	24,95	25,12	24,82	25,18	33,86	40,68	43,40	40,78	50,42	48,29
Predikce $ARIMA-G$	30,46	28,43	27,79	28,05	27,01	26,73	52,70	54,36	44,00	40,81	-3,78	52,42
Skutečnost	28,69	27,6	26,7	26	26,4	29,64	35,01	44,89	46,2	42,24	42,15	42,15

Hodina	61	62	63	64	65	66	67	68	69	70	71	72
Predikce $ARIMA$	34,19	31,70	42,70	37,12	37,27	57,14	59,82	51,27	47,06	35,92	32,15	25,91
Predikce $ARIMA-G$	33,93	32,15	42,70	42,92	37,27	57,14	59,82	51,27	43,96	37,37	33,33	28,42
Skutečnost	45,1	47,6	42,31	45,46	50	42	37	35,1	31	25,9	23,44	15,88

Tabulka 8: Predikce a skutečné hodnoty třídenní předpovědi MCP $ARIMA$ a $ARIMA - G$ pro 332. až 334. den (druhé největší zhoršení $ARIMA - G$).

Hodina	1	2	3	4	5	6	7	8	9	10	11	12
Predikce $ARIMA$	25,48	22,54	22,78	22,32	23,17	26,07	38,90	48,07	48,40	44,09	41,04	39,03
Predikce $ARIMA - G$	27,94	24,76	34,49	28,83	28,94	28,37	42,13	48,07	48,77	44,96	41,60	39,35
Test	18,99	16,99	15	14,76	15,6	23,24	35,5	39,94	41,9	42	40,51	42,44

Hodina	13	14	15	16	17	18	19	20	21	22	23	24
Predikce $ARIMA$	35,91	35,24	39,88	44,82	40,39	45,39	56,74	44,89	41,48	33,21	28,64	20,95
Predikce $ARIMA - G$	36,87	34,97	39,88	43,17	40,39	43,34	35,83	44,89	41,26	34,60	29,37	22,48
Test	53	55	55	55,01	59,42	67	64,9	59	55	45,71	41,4	34,58

Hodina	25	26	27	28	29	30	31	32	33	34	35	36
Predikce $ARIMA$	24,83	23,53	17,97	17,91	18,43	24,66	33,43	38,52	43,22	42,02	39,71	38,39
Predikce $ARIMA - G$	25,25	27,06	19,37	13,76	14,55	28,04	33,70	38,52	41,95	40,63	38,61	37,27
Test	35,25	34,61	34,24	35	36,53	38,24	58,3	80,6	85,1	82,3	80,8	81,5

Hodina	37	38	39	40	41	42	43	44	45	46	47	48
Predikce $ARIMA$	35,54	33,69	37,35	39,79	48,23	48,02	57,49	47,45	42,32	34,63	30,95	24,82
Predikce $ARIMA - G$	34,68	33,18	37,35	38,40	48,23	45,47	40,61	47,45	41,97	36,56	32,40	26,99
Test	79,63	81,96	81,87	82,13	93,01	110,1	105,1	88,3	77,1	61,28	46,91	39,01

Hodina	49	50	51	52	53	54	55	56	57	58	59	60
Predikce $ARIMA$	21,37	23,53	16,78	16,05	16,60	24,66	33,43	39,15	43,22	42,02	39,71	38,39
Predikce $ARIMA - G$	20,50	27,06	11,20	13,25	4,71	28,04	33,70	39,15	41,95	40,63	38,61	37,27
Test	43,9	41,81	39,5	38,09	39,5	43,63	61,08	89,4	99,8	96,5	94,99	93,97

Hodina	61	62	63	64	65	66	67	68	69	70	71	72
Predikce $ARIMA$	35,54	33,69	37,35	32,50	49,43	49,08	49,56	48,27	42,32	34,63	30,95	24,82
Predikce $ARIMA - G$	34,68	33,18	37,35	30,17	49,43	46,51	38,54	48,27	41,97	36,56	32,40	26,99
Test	87,8	88,1	89,15	91,2	98	116,12	106,35	89,2	73,4	58,05	40,62	39,01

Zdrojový kód

Kód 1: Úvod, vytvoření datasetů pro jednotlivé hodiny

```
# Nacteni knihoven a datasetu
install.packages("tseries")
install.packages('forecast', dependencies = TRUE)
install.packages("rugarch")
library(tseries)
library(forecast)
library(rugarch)
dataset<-read.csv2(file=file.path("source","mcp2_korigovany.csv"
    ))

#Rozdeleni puvodniho datasetu na 24 souboru dat jednotlivych
    hodin
{
  for(i in 1:24) {
    if(i < 10) {
      jmeno.dataset <- paste("h0", i, ".dataset", sep = "")
      jmeno.ts <- paste("h0", i, sep = "")
    }
    else{
      jmeno.dataset <- paste("h", i, ".dataset", sep = "")
      jmeno.ts <- paste("h", i, sep = "")
    }
    assign(jmeno.dataset,dataset[which(dataset$hodina==i),])
    assign(jmeno.ts, ts(dataset$mcp[which(dataset$hodina==i)],
       start=c(1),end=c(365)))
  }
}

#Q test
q.test.stata <- function (x) {
  Box.test(x, type="Ljung-Box", lag=min(length(x)/2-2, 40))
}

q.test.cipra <- function (x) {
  Box.test(x, type="Ljung-Box", lag=sqrt(length(x)))
}
```

Kód 2: Výzkum pro první hodinu (AIC, BIC)

```
for(j in 30:(365-7)) {
  for(i in (j-29):max(1,(j-199))) {
    #Vytvoreni prislusneho uciciho okna
    for.h1.train=window(h1,start=c(i),end=c(j))
    for.h1.test=window(h1, start=c(j+1), end=c(j+7))

    #p-hodnota ADF a KPSS testu pro vytvorene ucici okno
    adf=adf.test(for.h1.train,alternative="stationary")$p.value
    kpss=kpss.test(for.h1.train)$p.value

    #Vytvoreni modelu funkci auto.arima
    for.h1.adf.ai=auto.arima(for.h1.train, trace=FALSE, test="
       adf", ic="aic", stepwise = FALSE, approximation = TRUE)
    for.h1.adf.bi=auto.arima(for.h1.train, trace=FALSE, test="
       adf", ic="bic", stepwise = FALSE, approximation = TRUE)

    #Realizace predpovedi na 7 dni
    for.h1.adf.ai.f=forecast(for.h1.adf.ai, h=7)
    for.h1.adf.bi.f=forecast(for.h1.adf.bi, h=7)

    #Posouzeni presnosti sedmidenni predpovedi - vytvoreni
       datovych objektu
    acc.h1.adf.ai=accuracy(for.h1.adf.ai.f,for.h1.test)
    acc.h1.adf.bi=accuracy(for.h1.adf.bi.f,for.h1.test)

    #Zaznamenani parametru uciciho okna, extrakce hodnot z
       datovych objektu
    zacatek<-c(i)
    konec<-c(j)
    delka<-c(j-i+1)

    aic.chyba.rmse.training<-acc.h1.adf.ai[1,2]
    bic.chyba.rmse.training<-acc.h1.adf.bi[1,2]
    aic.chyba.rmse.test<-acc.h1.adf.ai[2,2]
    bic.chyba.rmse.test<-acc.h1.adf.bi[2,2]
    aic.chyba.mae.training<-acc.h1.adf.ai[1,3]
    bic.chyba.mae.training<-acc.h1.adf.bi[1,3]
    aic.chyba.mae.test<-acc.h1.adf.ai[2,3]
    bic.chyba.mae.test<-acc.h1.adf.bi[2,3]
    aic.chyba.mape.training<-acc.h1.adf.ai[1,5]
    bic.chyba.mape.training<-acc.h1.adf.bi[1,5]
```

```
      aic.chyba.mape.test<-acc.h1.adf.ai[2,5]
      bic.chyba.mape.test<-acc.h1.adf.bi[2,5]

      q.cipra.aic<-q.test.cipra(for.h1.adf.ai$residuals)$p.value
      q.stata.aic<-q.test.stata(for.h1.adf.ai$residuals)$p.value
      q.cipra.bic<-q.test.cipra(for.h1.adf.bi$residuals)$p.value
      q.stata.bic<-q.test.stata(for.h1.adf.bi$residuals)$p.value

      #Vytvoreni dataframu pro export
      if(i==1 && j==30) {
        predikce1=data.frame(zacatek, konec, delka, adf, kpss, q.
            cipra.aic, q.stata.aic, q.cipra.bic, q.stata.bic, aic.
            chyba.rmse.test, bic.chyba.rmse.test, aic.chyba.mae.
            test, bic.chyba.mae.test, aic.chyba.mape.test, bic.
            chyba.mape.test)
      }
      else {
        predikce1<-rbind(predikce1,c(zacatek, konec, delka, adf,
            kpss, q.cipra.aic, q.stata.aic, q.cipra.bic, q.stata.
            bic, aic.chyba.rmse.test, bic.chyba.rmse.test, aic.
            chyba.mae.test, bic.chyba.mae.test, aic.chyba.mape.test
            , bic.chyba.mape.test))
      }

      #Informacni tisk soucasne polohy do konzole
      coor<-c(i,j,delka)
      print(coor)
    }
}
```

Kód 3: Výzkum pro 24 souborů dat, uvedena 17 hodina

```
for(j in 50:(365-7)) {
  for(i in (j-49):max(1,(j-199))) {
    #Vytvoreni prislusneho uciciho okna
    for.train=window(h17,start=c(i),end=c(j))
    for.test=window(h17,start=c(j+1),end=c(j+7))

    #Stacionarita ADF&&KPSS
    #defaultni nastaveni p-hodnot vyssich diferenci na 0
    adf.d1=0
    kpss.d1=0
    adf.d2=0
    kpss.d2=0

    adf=adf.test(for.train,alternative="stationary")$p.value
    kpss=kpss.test(for.train)$p.value
    if(adf>0.05 || kpss<=0.05) {
      for.train.d1=diff(for.train)
      adf.d1=adf.test(for.train.d1,alternative="stationary")$p.
          value
      kpss.d1=kpss.test(for.train.d1)$p.value
      if(adf.d1>0.05 || kpss.d1<=0.05) {
        for.train.d2=diff(for.train.d1)
        adf.d2=adf.test(for.train.d2,alternative="stationary")$p
            .value
        kpss.d2=kpss.test(for.train.d2)$p.value
      }
    }

    #Arima modely dle diferencnich kriterii
    for.adf=auto.arima(for.train, trace=FALSE, test="adf", ic="
        aic", stepwise = TRUE, approximation = TRUE)
    for.kpss=auto.arima(for.train, trace=FALSE, test="kpss", ic=
        "aic", stepwise = TRUE, approximation = TRUE)

    #Realizace predpovedi pro 7 hodin z vytvorenych modelu
    for.adf.f=forecast(for.adf, h=7)
    for.kpss.f=forecast(for.kpss, h=7)

    #Posouzeni presnosti predpovedi
    acc.adf=accuracy(for.adf.f,for.test)
    acc.kpss=accuracy(for.kpss.f,for.test)
```

```
      #Postupne naplnovani dataframu - nejprve ucici okna
      zacatek<-c(i)
      konec<-c(j)
      delka<-c(j-i+1)

      #Chyby pro jednotliva ucici okna a diferencni kriteria
      adf.chyba.rmse.test<-acc.adf[2,2]
      kpss.chyba.rmse.test<-acc.kpss[2,2]

      adf.chyba.mae.test<-acc.adf[2,3]
      kpss.chyba.mae.test<-acc.kpss[2,3]

      adf.chyba.mape.test<-acc.adf[2,5]
      kpss.chyba.mape.test<-acc.kpss[2,5]

      #Overeni nezavislosti rezidui vytvorenych modelu
      adf.q.cipra<-q.test.cipra(for.adf$residuals)$p.value
      adf.q.stata<-q.test.stata(for.adf$residuals)$p.value
      kpss.q.cipra<-q.test.cipra(for.kpss$residuals)$p.value
      kpss.q.stata<-q.test.stata(for.kpss$residuals)$p.value

      #Vytvoreni dataframu pro export
      if(i==1 && j==50) {
        predikcefor=data.frame(zacatek, konec, delka, adf, kpss,
           adf.d1, kpss.d1, adf.d2, kpss.d2, adf.q.cipra, adf.q.
           stata, kpss.q.cipra, kpss.q.stata, adf.chyba.rmse.test,
            kpss.chyba.rmse.test, adf.chyba.mae.test, kpss.chyba.
           mae.test, adf.chyba.mape.test, kpss.chyba.mape.test)
      }
      else {
        predikcefor<-rbind(predikcefor,c(zacatek, konec, delka,
           adf, kpss, adf.d1, kpss.d1, adf.d2, kpss.d2, adf.q.
           cipra, adf.q.stata, kpss.q.cipra, kpss.q.stata, adf.
           chyba.rmse.test, kpss.chyba.rmse.test, adf.chyba.mae.
           test, kpss.chyba.mae.test, adf.chyba.mape.test, kpss.
           chyba.mape.test))
      }
      #Informacni tisk soucasne polohy do konzole
      coor<-c(i,j,delka)
      print(coor)
    }
  }
```

```
write.csv2(predikcefor, file="hodiny_17.csv")
remove(predikcefor)
```

Kód 4: Export grafů z výzkumu pro první hodinu, analýza rozdílu maximální *RMSE* pro *AIC* a *BIC*

```
#Export informaci z puvodniho pruzkumu prvni hodiny BIC AIC
    30-200
{
  current.dataset<-read.csv2(file=file.path("hodina_1_aic_bic.
      csv"))
  h.cislo.hodiny <- paste("h01_puvodni", sep = "")

  #Maximalni RMSE na delku uceni
  for(i in min(current.dataset$delka):max(current.dataset$delka)
      ) {
    rmse.max.set=current.dataset[current.dataset$delka==i,]
    rmse.max.aic=max(rmse.max.set$aic.chyba.rmse.test)
    rmse.max.bic=max(rmse.max.set$bic.chyba.rmse.test)
    delka<-i
    if(i==min(current.dataset$delka)) {
      rmse.max.length=data.frame(delka,rmse.max.aic,rmse.max.bic
          )
    }
    else {
      rmse.max.length<-rbind(rmse.max.length,c(delka,rmse.max.
          aic,rmse.max.bic))
    }
  }

  #Export - plot zavislosti maximalni RMSE kriteriem AIC na
      delce uceni
  png(filename=paste(h.cislo.hodiny,"_length_rmse_max_aic.png",
      sep=""))
  plot(x<-rmse.max.length$delka,y<-rmse.max.length$rmse.max.aic,
      type="h", ylim=c(0,100), cex.lab=1.3, cex.axis=1.3, cex.
      main=1.3, cex.sub=1.3, main = paste("Hodina ",1,sep = ""),
      xlab="Delka uciciho obdobi",ylab="Max RMSE (AIC)")
  dev.off()

  #Export - plot zavislosti maximalni RMSE kriteriem BIC na
      delce uceni
  png(filename=paste(h.cislo.hodiny,"_length_rmse_max_bic.png",
      sep=""))
```

```
plot(x<-rmse.max.length$delka,y<-rmse.max.length$rmse.max.bic,
    type="h", ylim=c(0,100), cex.lab=1.3, cex.axis=1.3, cex.
    main=1.3, cex.sub=1.3, main = paste("Hodina ",1,sep = ""),
    xlab="Delka uciciho obdobi",ylab="Max RMSE (BIC)")
dev.off()


#Minimalni RMSE na delku uceni
for(i in min(current.dataset$delka):max(current.dataset$delka)
    ) {
  rmse.min.set=current.dataset[current.dataset$delka==i,]
  rmse.min.aic=min(rmse.min.set$aic.chyba.rmse.test)
  rmse.min.bic=min(rmse.min.set$bic.chyba.rmse.test)
  delka<-i
  if(i==min(current.dataset$delka)) {
    rmse.min.length=data.frame(delka,rmse.min.aic,rmse.min.bic
        )
  }
  else {
    rmse.min.length<-rbind(rmse.min.length,c(delka,rmse.min.
        aic,rmse.min.bic))
  }
}


#Export - plot zavislosti minimalni RMSE kriteriem AIC na
    delce uceni
png(filename=paste(h.cislo.hodiny,"_length_rmse_min_aic.png",
    sep=""))
plot(x<-rmse.min.length$delka,y<-rmse.min.length$rmse.min.aic,
    type="h", ylim=c(0,6), cex.lab=1.3, cex.axis=1.3, cex.main
    =1.3, cex.sub=1.3, main = paste("Hodina ",1,sep = ""),xlab=
    "Delka uciciho obdobi",ylab="min RMSE (AIC)")
dev.off()

#Export - plot zavislosti minimalni RMSE kriteriem BIC na
    delce uceni
png(filename=paste(h.cislo.hodiny,"_length_rmse_min_bic.png",
    sep=""))
plot(x<-rmse.min.length$delka,y<-rmse.min.length$rmse.min.bic,
    type="h", ylim=c(0,6), cex.lab=1.3, cex.axis=1.3, cex.main
    =1.3, cex.sub=1.3, main = paste("Hodina ",1,sep = ""),xlab=
    "Delka uciciho obdobi",ylab="min RMSE (BIC)")
```

```
dev.off()

#Minimalni RMSE v zavislosti na zacatku predpovedniho obdobi
for(i in min(current.dataset$konec):max(current.dataset$konec)
    ) {
  rmse.pos.min.set=current.dataset[current.dataset$konec==i,]
  rmse.pos.min.aic=min(rmse.pos.min.set$aic.chyba.rmse.test)
  rmse.pos.min.bic=min(rmse.pos.min.set$bic.chyba.rmse.test)
  zacatek<-i+1
  if(i==min(current.dataset$konec)) {
    rmse.min.pos=data.frame(zacatek,rmse.pos.min.aic,rmse.pos.
        min.bic)
  }
  else {
    rmse.min.pos<-rbind(rmse.min.pos,c(zacatek,rmse.pos.min.
        aic,rmse.pos.min.bic))
  }
}

View(rmse.min.pos)

#Export - plot zavislosti minimalni RMSE testem AIC na zacatku
    predpovedi
png(filename=paste(h.cislo.hodiny,"_position_rmse_min_aic.png"
    ,sep=""))
plot(x<-rmse.min.pos$zacatek,y<-rmse.min.pos$rmse.pos.min.aic,
    type="h", ylim=c(0,40), main = paste("Hodina ",1,sep = ""),
     cex.lab=1.3, cex.axis=1.3, cex.main=1.3, cex.sub=1.3, xlab
    ="Zacatek predpovedi",ylab="Min RMSE (AIC)")
dev.off()

#Export - plot zavislosti minimalni RMSE testem BIC na zacatku
    predpovedi
png(filename=paste(h.cislo.hodiny,"_position_rmse_min_bic.png"
    ,sep=""))
plot(x<-rmse.min.pos$zacatek,y<-rmse.min.pos$rmse.pos.min.bic,
    type="h", ylim=c(0,40), main = paste("Hodina ",1,sep = ""),
     cex.lab=1.3, cex.axis=1.3, cex.main=1.3, cex.sub=1.3, xlab
    ="Zacatek predpovedi",ylab="Min RMSE (BIC)")
dev.off()


#Prumerna RMSE na delku uceni
```

```
for(i in min(current.dataset$delka):max(current.dataset$delka)
    ) {
  rmse.mean.set=current.dataset[current.dataset$delka==i,]
  rmse.mean.aic=mean(rmse.mean.set$aic.chyba.rmse.test)
  rmse.mean.bic=mean(rmse.mean.set$bic.chyba.rmse.test)
  delka<-i
  if(i==min(current.dataset$delka)) {
    rmse.mean.length=data.frame(delka,rmse.mean.aic,rmse.mean.
        bic)
  }
  else {
    rmse.mean.length<-rbind(rmse.mean.length,c(delka,rmse.mean
        .aic,rmse.mean.bic))
  }
}

#Export - plot zavislosti prumerne RMSE testem AIC na delce
    uceni
png(filename=paste(h.cislo.hodiny,"_length_rmse_mean_aic.png",
    sep=""))
plot(x<-rmse.mean.length$delka,y<-rmse.mean.length$rmse.mean.
    aic,type="h", ylim=c(0,20), main = paste("Hodina ",1,sep =
    ""),xlab="Delka uciciho obdobi",ylab="Prumerna RMSE (AIC)",
    cex.lab=1.3, cex.axis=1.3, cex.main=1.3, cex.sub=1.3)
dev.off()

#Export - plot zavislosti prumerne RMSE testem BIC na delce
    uceni
png(filename=paste(h.cislo.hodiny,"_length_rmse_mean_bic.png",
    sep=""))
plot(x<-rmse.mean.length$delka,y<-rmse.mean.length$rmse.mean.
    bic,type="h", ylim=c(0,20), main = paste("Hodina ",1,sep =
    ""),xlab="Delka uciciho obdobi",ylab="Prumerna RMSE (BIC)",
    cex.lab=1.3, cex.axis=1.3, cex.main=1.3, cex.sub=1.3)
dev.off()



#Minimalni p-hodnota Q-statstiky v zavislosti na delce uceni
for(i in min(current.dataset$delka):max(current.dataset$delka)
    ) {
  q.min.set=current.dataset[current.dataset$delka==i,]
  q.min.aic=min(q.min.set$q.cipra.aic)
```

```
  q.min.bic=min(q.min.set$q.cipra.bic)
  delka<-i
  if(i==min(current.dataset$delka)) {
    q.min.length=data.frame(delka,q.min.aic,q.min.bic)
  }
  else {
    q.min.length<-rbind(q.min.length,c(delka,q.min.aic,q.min.
        bic))
  }
}

#Export - plot zavislosti minimalni p-hodnoty Q-statstiky
    testem AIC na delce uceni
png(filename=paste(h.cislo.hodiny,"_length_q_min_aic.png",sep=
    ""))
plot(x<-q.min.length$delka,y<-q.min.length$q.min.aic,type="h",
     ylim=c(0,1), main = paste("Hodina ",1,sep = ""),xlab="
    Delka uciciho obdobi",ylab="Minimalni p-hod. Q stat. (AIC)"
    ,cex.lab=1.3, cex.axis=1.3, cex.main=1.3, cex.sub=1.3)
abline(h=0.05, col="blue")
dev.off()

#Export - plot zavislosti minimalni p-hodnoty Q-statstiky
    testem BIC na delce uceni
png(filename=paste(h.cislo.hodiny,"_length_q_min_bic.png",sep=
    ""))
plot(x<-q.min.length$delka,y<-q.min.length$q.min.bic,type="h",
     ylim=c(0,1), main = paste("Hodina ",1,sep = ""),xlab=Delka
     uciciho obdobi",ylab="Minimalni p-hod. Q stat. (BIC)",cex.
    lab=1.3, cex.axis=1.3, cex.main=1.3, cex.sub=1.3)
abline(h=0.05, col="blue")
dev.off()

#Export - statistiky AIC versus BIC
min.mean.delka.rmse.aic <- rmse.mean.length$delka[rmse.mean.
    length$rmse.mean.aic==min(rmse.mean.length$rmse.mean.aic)]
min.mean.delka.rmse.bic <- rmse.mean.length$delka[rmse.mean.
    length$rmse.mean.bic==min(rmse.mean.length$rmse.mean.bic)]
min.mean.hodnota.rmse.aic <- rmse.mean.length$rmse.mean.aic[
    rmse.mean.length$rmse.mean.aic==min(rmse.mean.length$rmse.
    mean.aic)]
min.mean.hodnota.rmse.bic <- rmse.mean.length$rmse.mean.bic[
    rmse.mean.length$rmse.mean.bic==min(rmse.mean.length$rmse.
```

```
    mean.bic)]

mean.rmse.aic <- mean(current.dataset$aic.chyba.rmse.test)
mean.rmse.bic <- mean(current.dataset$bic.chyba.rmse.test)
delka.datasetu <- length(current.dataset)

rmse.bic.minus.aic=mean(current.dataset$bic.chyba.rmse.test)-
    mean(current.dataset$aic.chyba.rmse.test)
mae.bic.minus.aic=mean(current.dataset$bic.chyba.mae.test)-
    mean(current.dataset$aic.chyba.mae.test)
mape.bic.minus.aic=mean(current.dataset$bic.chyba.mape.test)-
    mean(current.dataset$aic.chyba.mape.test)

aic.presnejsi.rmse=0
bic.presnejsi.rmse=0

for(i in 1:length(current.dataset$delka)) {

  if(current.dataset$bic.chyba.rmse.test[i] > current.dataset$
      aic.chyba.rmse.test[i]) {
    aic.presnejsi.rmse <- aic.presnejsi.rmse + 1
  }

  if(current.dataset$bic.chyba.rmse.test[i] < current.dataset$
      aic.chyba.rmse.test[i]) {
    bic.presnejsi.rmse <- bic.presnejsi.rmse + 1
  }
}

h1.statistika=data.frame(min.mean.delka.rmse.aic, min.mean.
    delka.rmse.bic, min.mean.hodnota.rmse.aic, min.mean.hodnota
    .rmse.bic, mean.rmse.aic, mean.rmse.bic, rmse.bic.minus.aic
    , mae.bic.minus.aic, mape.bic.minus.aic, aic.presnejsi.rmse
    , bic.presnejsi.rmse, delka.datasetu)
write.csv2(h1.statistika, file="h1_statistika.csv")

#Statistiky 'Lepsi o pul' v zavislosti maximalni RMSE na delce
    uciciho obdobi
{
lepsi.o.pul=0
for(i in 1:length(rmse.max.length$rmse.max.aic)) {
  if(rmse.max.length$rmse.max.aic[i]<=rmse.max.length$rmse.max
      .bic[i]/2) {
```

```
      lepsi.o.pul <- lepsi.o.pul + 1
      delka<-rmse.max.length$delka[i]
      if(lepsi.o.pul == 1) {
        souradnice=data.frame(delka)
      }
      else {
        souradnice<-rbind(souradnice,c(delka))
      }
    }
  }

  write.csv2(souradnice, file="lepsi_o_pul.csv")
  }
}
```

Kód 5: Export grafů z výzkumu pro 24 datových souborů

```
for(h in 1:24) {
  current.dataset<-read.csv2(file=file.path(paste("hodiny_",h,".
     csv",sep="")))

  if(h < 10) {
    h.cislo.hodiny <- paste("h0", h, sep = "")
  }
  else{

    h.cislo.hodiny <- paste("h", h, sep = "")
  }

  #Maximalni RMSE na delku uceni
  for(i in min(current.dataset$delka):max(current.dataset$delka)
     ) {
    rmse.max.set=current.dataset[current.dataset$delka==i,]
    rmse.max.adf=max(rmse.max.set$adf.chyba.rmse.test)
    rmse.max.kpss=max(rmse.max.set$kpss.chyba.rmse.test)
    delka<-i
    if(i==min(current.dataset$delka)) {
      rmse.max.length=data.frame(delka,rmse.max.adf,rmse.max.
         kpss)
    }
    else {
      rmse.max.length<-rbind(rmse.max.length,c(delka,rmse.max.
         adf,rmse.max.kpss))
    }
```

```
}



#Export - plot zavislosti maximalni RMSE testem ADF na delce
    uceni
png(filename=paste(h.cislo.hodiny,"_length_rmse_max_adf.png",
    sep=""))
plot(x<-rmse.max.length$delka,y<-rmse.max.length$rmse.max.adf,
    type="h", ylim=c(0,100), main = paste("Hodina ",h,sep = "")
    ,xlab="Delka uciciho obdobi'',ylab="Max RMSE (ADF)", cex.
    lab=1.3, cex.axis=1.3, cex.main=1.3, cex.sub=1.3)
dev.off()

#Export - plot zavislosti maximalni RMSE testem KPSS na delce
    uceni
png(filename=paste(h.cislo.hodiny,"_length_rmse_max_kpss.png",
    sep=""))
plot(x<-rmse.max.length$delka,y<-rmse.max.length$rmse.max.kpss
    ,type="h", ylim=c(0,100), main = paste("Hodina ",h,sep = ""
    ),xlab="Delka uciciho obdobi'",ylab="Max RMSE (KPSS)", cex.
    lab=1.3, cex.axis=1.3, cex.main=1.3, cex.sub=1.3)
dev.off()

#Minimalni RMSE na delku uceni
for(i in min(current.dataset$delka):max(current.dataset$delka)
    ) {
  rmse.min.set=current.dataset[current.dataset$delka==i,]
  rmse.min.adf=min(rmse.min.set$adf.chyba.rmse.test)
  rmse.min.kpss=min(rmse.min.set$kpss.chyba.rmse.test)
  delka<-i
  if(i==min(current.dataset$delka)) {
    rmse.min.length=data.frame(delka,rmse.min.adf,rmse.min.
        kpss)
  }
  else {
    rmse.min.length<-rbind(rmse.min.length,c(delka,rmse.min.
        adf,rmse.min.kpss))
  }
}

#Export - plot zavislosti minimalni RMSE testem ADF na delce
    uceni
```

```
png(filename=paste(h.cislo.hodiny,"_length_rmse_min_adf.png",
    sep=""))
plot(x<-rmse.min.length$delka,y<-rmse.min.length$rmse.min.adf,
    type="h", ylim=c(0,6), main = paste("Hodina ",h,sep = ""),
    xlab="Delka uciciho obdobi'",ylab="Min RMSE (ADF)", cex.lab
    =1.3, cex.axis=1.3, cex.main=1.3, cex.sub=1.3)
dev.off()

#Export - plot zavislosti minimalni RMSE testem KPSS na delce
    uceni
png(filename=paste(h.cislo.hodiny,"_length_rmse_min_kpss.png",
    sep=""))
plot(x<-rmse.min.length$delka,y<-rmse.min.length$rmse.min.kpss
    ,type="h", ylim=c(0,6), main = paste("Hodina ",h,sep = ""),
    xlab="Delka uciciho obdobi'",ylab="Min RMSE (KPSS)", cex.
    lab=1.3, cex.axis=1.3, cex.main=1.3, cex.sub=1.3)
dev.off()

#Minimalni RMSE v zavislosti na zacatku predpovedniho obdobi
for(i in min(current.dataset$konec):max(current.dataset$konec)
    ) {
  rmse.pos.min.set=current.dataset[current.dataset$konec==i,]
  rmse.pos.min.adf=min(rmse.pos.min.set$adf.chyba.rmse.test)
  rmse.pos.min.kpss=min(rmse.pos.min.set$kpss.chyba.rmse.test)
  zacatek<-i+1
  if(i==min(current.dataset$konec)) {
    rmse.min.pos=data.frame(zacatek,rmse.pos.min.adf,rmse.pos.
        min.kpss)
  }
  else {
    rmse.min.pos<-rbind(rmse.min.pos,c(zacatek,rmse.pos.min.
        adf,rmse.pos.min.kpss))
  }
}

#Export - plot zavislosti minimalni RMSE testem ADF na delce
    uceni
png(filename=paste(h.cislo.hodiny,"_position_rmse_min_adf.png
    ",sep=""))
plot(x<-rmse.min.pos$zacatek,y<-rmse.min.pos$rmse.pos.min.adf,
    type="h", ylim=c(0,40), main = paste("Hodina ",h,sep = ""),
     xlab="Zacatek predpovedi" ,ylab="Min RMSE (ADF)", cex.lab
    =1.3, cex.axis=1.3, cex.main=1.3, cex.sub=1.3)
```

```
dev.off()

#Export - plot zavislosti minimalni RMSE testem KPSS na delce
    uceni
png(filename=paste(h.cislo.hodiny,"_position_rmse_min_kpss.png
    ",sep=""))
plot(x<-rmse.min.pos$zacatek,y<-rmse.min.pos$rmse.pos.min.kpss
    ,type="h", ylim=c(0,40), main = paste("Hodina ",h,sep = "")
    ,xlab="Zacatek predpovedi",ylab="Min RMSE (KPSS)", cex.lab
    =1.3, cex.axis=1.3, cex.main=1.3, cex.sub=1.3)
dev.off()

#Prumerna RMSE na delku uceni
for(i in min(current.dataset$delka):max(current.dataset$delka)
    ) {
  rmse.mean.set=current.dataset[current.dataset$delka==i,]
  rmse.mean.adf=mean(rmse.mean.set$adf.chyba.rmse.test)
  rmse.mean.kpss=mean(rmse.mean.set$kpss.chyba.rmse.test)
  delka<-i
  if(i==min(current.dataset$delka)) {
    rmse.mean.length=data.frame(delka,rmse.mean.adf,rmse.mean.
        kpss)
  }
  else {
    rmse.mean.length<-rbind(rmse.mean.length,c(delka,rmse.mean
        .adf,rmse.mean.kpss))
  }
}

#Export - plot zavislosti prumerne RMSE testem ADF na delce
    uceni
png(filename=paste(h.cislo.hodiny,"_length_rmse_mean_adf.png",
    sep=""))
plot(x<-rmse.mean.length$delka,y<-rmse.mean.length$rmse.mean.
    adf,type="h", ylim=c(0,20), main = paste("Hodina ",h,sep =
    ""),xlab="Delka uciciho obdobi'",ylab="Prumerna RMSE (ADF)"
    , cex.lab=1.3, cex.axis=1.3, cex.main=1.3, cex.sub=1.3)
dev.off()

#Export - plot zavislosti prumerne RMSE testem KPSS na delce
    uceni
png(filename=paste(h.cislo.hodiny,"_length_rmse_mean_kpss.png"
    ,sep=""))
```

```
plot(x<-rmse.mean.length$delka,y<-rmse.mean.length$rmse.mean.
    kpss,type="h", ylim=c(0,20), main = paste("Hodina ",h,sep =
     ""),xlab="Delka uciciho obdobi'",ylab="Prumerna RMSE (KPSS
    )", cex.lab=1.3, cex.axis=1.3, cex.main=1.3, cex.sub=1.3)
dev.off()

#Minimalni p-hodnota Q-statstiky v zavislosti na delce uceni
for(i in min(current.dataset$delka):max(current.dataset$delka)
    ) {
  q.min.set=current.dataset[current.dataset$delka==i,]
  q.min.adf=mean(q.min.set$adf.q.cipra)
  q.min.kpss=mean(q.min.set$kpss.q.cipra)
  delka<-i
  if(i==min(current.dataset$delka)) {
    q.min.length=data.frame(delka,q.min.adf,q.min.kpss)
  }
  else {
    q.min.length<-rbind(q.min.length,c(delka,q.min.adf,q.min.
        kpss))
  }
}

#Export - plot zavislosti minimalni p-hodnoty Q-statstiky
    testem ADF na delce uceni
png(filename=paste(h.cislo.hodiny,"_length_q_min_adf.png",sep
    =""))
plot(x<-q.min.length$delka,y<-q.min.length$q.min.adf,type="h",
     ylim=c(0,1), main = paste("Hodina ",h,sep = ""),xlab="
    Delka uciciho obdobi",ylab="Minimalni p-hod. Q stat. (ADF)
    ", cex.lab=1.3, cex.axis=1.3, cex.main=1.3, cex.sub=1.3)
abline(h=0.05, col="blue")
dev.off()

#Export - plot zavislosti minimalni p-hodnoty Q-statstiky
    testem KPSS na delce uceni
png(filename=paste(h.cislo.hodiny,"_length_q_min_kpss.png",sep
    =""))
plot(x<-q.min.length$delka,y<-q.min.length$q.min.kpss,type="h
    ", ylim=c(0,1), main = paste("Hodina ",h,sep = ""),xlab=
    Delka uciciho obdobi",ylab="Minimalni p-hod. Q stat. (KPSS)
    ", cex.lab=1.3, cex.axis=1.3, cex.main=1.3, cex.sub=1.3)
abline(h=0.05, col="blue")
dev.off()
```

```
#Pocet diferencovani v zavislosti na delce uceni
for(i in min(current.dataset$delka):max(current.dataset$delka)
    ) {
  dif.count.set=current.dataset[current.dataset$delka==i,]

  pocet.dif0<-length(dif.count.set$adf[dif.count.set$adf.d1 ==
      0])

  pocet.dif2<-length(dif.count.set$adf.d2[dif.count.set$adf.d2
      != 0])

  pocet.dif1<-length(dif.count.set$adf.d1[dif.count.set$adf.d1
      != 0]) - pocet.dif2

  delka<-i
  if(i==min(current.dataset$delka)) {
    dif.count.length=data.frame(delka,pocet.dif0,pocet.dif1,
       pocet.dif2)
  }
  else {
    dif.count.length<-rbind(dif.count.length,c(delka,pocet.
       dif0,pocet.dif1,pocet.dif2))
  }
}

#Export - plot zavislosti poctu diferencovani 2 na delce uceni
png(filename=paste(h.cislo.hodiny,"_length_dif2_count.png",sep
    =""))
plot(x<-dif.count.length$delka,y<-dif.count.length$pocet.dif2,
    type="h", ylim=c(0,20), main = paste("Hodina ",h,sep = ""),
    xlab="Delka uciciho obdobi'",ylab="Pocet 2. diferencovani",
     cex.lab=1.3, cex.axis=1.3, cex.main=1.3, cex.sub=1.3)
dev.off()

#Export - plot zavislosti poctu diferencovani 1 na delce uceni
png(filename=paste(h.cislo.hodiny,"_length_dif1_count.png",sep
    =""))
plot(x<-dif.count.length$delka,y<-dif.count.length$pocet.dif1,
    type="h", ylim=c(0,240), main = paste("Hodina ",h,sep = "")
    ,xlab="Delka uciciho obdobi'",ylab="Pocet 1. diferencovani
    ", cex.lab=1.3, cex.axis=1.3, cex.main=1.3, cex.sub=1.3)
dev.off()
```

```
#Export - plot zavislosti poctu diferencovani 0 na delce uceni
png(filename=paste(h.cislo.hodiny,"_length_dif0_count.png",sep
    =""))
plot(x<-dif.count.length$delka,y<-dif.count.length$pocet.dif0,
    type="h", ylim=c(0,260), main = paste("Hodina ",h,sep = "")
    ,xlab="Delka uciciho obdobi'",ylab="Pocet bez diferencovani
    ", cex.lab=1.3, cex.axis=1.3, cex.main=1.3, cex.sub=1.3)
dev.off()

#Hledani a export nejmensi prumerne chyby pro danou hodinu
min.mean.rmse.adf <- rmse.mean.length$delka[rmse.mean.length$
    rmse.mean.adf==min(rmse.mean.length$rmse.mean.adf)]
min.mean.rmse.kpss <- rmse.mean.length$delka[rmse.mean.length$
    rmse.mean.kpss==min(rmse.mean.length$rmse.mean.kpss)]

kpss.nedif.adf.ano=0
adf.pomohlo=0
kpss.nedif.adf.ano.d1=0
adf.pomohlo.d1=0
adf.presnejsi.rmse=0
kpss.presnejsi.rmse=0

for(i in 1:length(current.dataset$adf)) {
  if(current.dataset$adf[i]>0.05 && current.dataset$kpss[i
     ]>0.05) {
    kpss.nedif.adf.ano <- kpss.nedif.adf.ano + 1
    if(current.dataset$adf.chyba.rmse.test[i] < current.
       dataset$kpss.chyba.rmse.test[i]) {
      adf.pomohlo <- adf.pomohlo + 1
    }
  }

  if(current.dataset$adf.d1[i]>0.05 && current.dataset$kpss.d1
     [i]>0.05) {
    kpss.nedif.adf.ano.d1 <- kpss.nedif.adf.ano.d1 + 1
    if(current.dataset$adf.chyba.rmse.test[i] < current.
       dataset$kpss.chyba.rmse.test[i]) {
      adf.pomohlo.d1 <- adf.pomohlo.d1 + 1
    }
  }

  if(current.dataset$kpss.chyba.rmse.test[i] > current.dataset
```

```
        $adf.chyba.rmse.test[i]) {
        adf.presnejsi.rmse <- adf.presnejsi.rmse + 1
      }

      if(current.dataset$kpss.chyba.rmse.test[i] < current.dataset
        $adf.chyba.rmse.test[i]) {
        kpss.presnejsi.rmse <- kpss.presnejsi.rmse + 1
      }
    }

    if(h==1) {
      nejmensi.prumerne.chyby.delky=data.frame(h, min.mean.rmse.
        adf, min.mean.rmse.kpss, kpss.nedif.adf.ano, adf.pomohlo,
         kpss.nedif.adf.ano.d1, adf.pomohlo.d1, adf.presnejsi.
        rmse, kpss.presnejsi.rmse)
    }
    else {
      nejmensi.prumerne.chyby.delky<-rbind(nejmensi.prumerne.chyby
        .delky,c(h, min.mean.rmse.adf, min.mean.rmse.kpss, kpss.
        nedif.adf.ano, adf.pomohlo, kpss.nedif.adf.ano.d1, adf.
        pomohlo.d1, adf.presnejsi.rmse, kpss.presnejsi.rmse))
    }

    print(h)

}
```

Kód 6: Statistika *RMSE* výzkumu pro 24 hodin a graf průměrné *RMSE* při nejlepší učící délce v jednotlivých hodinách

```
analyza.24 <- read.csv2(file=file.path("nejmensi_prumerne_chyby_
    delky.csv"))

#Vytvoreni grafu RMSE v zavislosti na hodine
png(filename="analyza_24_RMSE(hodina).png")
plot(x<-analyza.24$h, y<-analyza.24$min.mean.hodnota.adf, type="
    l", col="red", main = "RMSE v jednotlivych hodinach", ylim=c
    (0,20), xlab="hodina", ylab="RMSE", axes = FALSE)
axis(side = 1, at = c
    (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24)
    )
axis(side = 2, at = c(5,10,15,20))
lines(y<-analyza.24$min.mean.hodnota.kpss, type="l", col="black"
    , axes = FALSE)
```

```
dev.off()

plot(x<-analyza.24$h, y<-analyza.24$min.mean.hodnota.adf, type="
    l", col="red", ylim=c(0,20), xlab="hodina", ylab="RMSE", axes
     = FALSE)
axis(side = 1, at = c
    (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24)
    )
axis(side = 2, at = c(5,10,15,20))
lines(y<-analyza.24$min.mean.hodnota.kpss, type="l", col="black"
    , axes = FALSE)

#Prurezove statstiky pro testy KPSS a ADF
max.adf<-max(analyza.24$min.mean.hodnota.adf)
min.adf<-min(analyza.24$min.mean.hodnota.adf)
mean.adf<-mean(analyza.24$min.mean.hodnota.adf)
sd.adf<-sd(analyza.24$min.mean.hodnota.adf)

max.kpss<-max(analyza.24$min.mean.hodnota.kpss)
min.kpss<-min(analyza.24$min.mean.hodnota.kpss)
mean.kpss<-mean(analyza.24$min.mean.hodnota.kpss)
sd.kpss<-sd(analyza.24$min.mean.hodnota.kpss)

statstiky.24<-data.frame(max.adf,min.adf,mean.adf,sd.adf,max.
    kpss,min.kpss,mean.kpss,sd.kpss)
write.csv2(statstiky.24, file="statstiky24_sd.csv")
```

Kód 7: Výpočet chyb, předpovědní funkce, vykreslovací funkce

```
#Funkce pro vypocet RMSE
pocitej.rmse <- function (point.vektor.predpoved, point.vektor.
    test) {
  for(i in 1:length(point.vektor.predpoved)) {
    if(i == 1) {
      SE.suma.ctvercu<-(point.vektor.predpoved[i]-point.vektor.
          test[i])^2
    }
    else{
      SE.suma.ctvercu<-SE.suma.ctvercu+(point.vektor.predpoved[i
          ]-point.vektor.test[i])^2
    }
  }
  SE.prumer<-SE.suma.ctvercu/length(point.vektor.predpoved)
  RMSE<-sqrt(SE.prumer)
```

```
  return(RMSE)
}

#Funkce pro vypocet MAE
pocitej.mae <- function (point.vektor.predpoved, point.vektor.
    test) {
  for(i in 1:length(point.vektor.predpoved)) {
    if(i == 1) {
      AE.suma.abs<-abs(point.vektor.predpoved[i]-point.vektor.
          test[i])
    }
    else{
      AE.suma.abs<-AE.suma.abs+abs(point.vektor.predpoved[i]-
          point.vektor.test[i])
    }
  }
  MAE<-AE.suma.abs/length(point.vektor.predpoved)
  return(MAE)
}

#Funkce pro vypocet MAPE
pocitej.mape <- function (point.vektor.predpoved, point.vektor.
    test) {
  for(i in 1:length(point.vektor.predpoved)) {
    if(i == 1) {
      APE.suma.abs<-abs(point.vektor.predpoved[i]-point.vektor.
          test[i])/point.vektor.test[i]
    }
    else{
      APE.suma.abs<-APE.suma.abs+abs(point.vektor.predpoved[i]-
          point.vektor.test[i])/point.vektor.test[i]
    }
  }
  MAPE<-APE.suma.abs/length(point.vektor.predpoved)
  return(MAPE)
}


#Predpovedni funkce kompatibilni s funkci plot pro vykreslovani
predpovidej.plot <- function (den,delka) {

  #Nacteni optimalnich ucicich delek
  settings<-read.csv2(file=file.path("source","settings.csv"))
```

```
#Pro kazdou z 24 hodin
for(h in 1:24) {
  #Nastaveni hodiny v predpovednim algoritmu
  if(h < 10) {
    h.cislo.hodiny <- paste("h0", h, sep = "")
    jmeno.predpovedi <- paste("f0", h, sep = "")
  }
  else{
    h.cislo.hodiny <- paste("h", h, sep = "")
    jmeno.predpovedi <- paste("f", h, sep = "")
  }


  # Jestli je chyba KPSS pro jeho nejlepsi ucebni delku mensi
     nez u ADF, predpovidej KPSS
  if(settings$min.mean.hodnota.kpss[h] <= settings$min.mean.
     hodnota.adf[h]) {
    for.train=window(get(h.cislo.hodiny),start=c(den-settings$
       min.mean.rmse.kpss[h]),end=c(den-1))
    for.model=auto.arima(for.train, trace=FALSE, test="kpss",
       ic="aic", stepwise = FALSE, approximation = FALSE)
    print(paste("h ", h, " kpss", sep = ""))
  }
  # Jinak predpovidej ADF
  else {
    for.train=window(get(h.cislo.hodiny),start=c(den-settings$
       min.mean.rmse.adf[h]),end=c(den-1))
    for.model=auto.arima(for.train, trace=FALSE, test="adf",
       ic="aic", stepwise = FALSE, approximation = FALSE)
    print(paste("h ", h, " adf", sep = ""))
  }

  #Vytvoreni predpovedi na zaklade sestrojeneho modelu a
     extrakce jejich hodnot
  for.forecast=forecast(for.model, h=delka)
  assign(jmeno.predpovedi, for.forecast$mean, envir = parent.
     frame())
  print(jmeno.predpovedi)
  print(for.forecast$mean)

  #Extrakce parametru modelu
  koeficienty<-for.model$coef
```

```
    koeficienty_jmena<-names(koeficienty)
    koeficienty_cisla<-unname(koeficienty)

    sigma2<-round(for.model$sigma2,3)
    q.rezidua<-round(q.test.cipra(for.model$residuals)$p.value
        ,3)
    rad.arima<-paste("(",toString(arimaorder(for.model)),")",sep
        = "")
    koef.jmena<-toString(koeficienty_jmena)
    koef.cisla<-toString(round(koeficienty_cisla,3))

    #Ulozeni parametru ARIMA modelu
    options(stringsAsFactors = FALSE)

    if (h==1) {
      parametry.modelu=data.frame(h,sigma2,q.rezidua,rad.arima,
          koef.jmena,koef.cisla)
    }
    else {
      parametry.modelu<-rbind(parametry.modelu,c(h,sigma2,q.
          rezidua,rad.arima,koef.jmena,koef.cisla))

    }

}

#Vytvoreni tabulky predpovedi
assign(paste("parametry.modelu",sep=""), parametry.modelu,
    envir = parent.frame())

forecast.table.method<-data.frame(f01,f02,f03,f04,f05,f06,f07,
    f08,f09,f10,f11,
                                      f12,f13,f14,f15,f16,f17,f18,
                                          f19,f20,f21,f22,f23,f24)

assign(paste("forecast.table",sep=""), forecast.table.method,
    envir = parent.frame())

for (i in 1:delka) {
  if(i == 1) {
    predpoved.point.vektor <- forecast.table[i,]
  }
  else {
```

```
      predpoved.point.vektor <- c(predpoved.point.vektor,
         forecast.table[i,])
    }
  }
  #Tisk ziskanych dat do konzole a extrakce do globalnich
     promennych
  print(predpoved.point.vektor)
  assign(paste("point.vektor.predpoved.arima",sep=""), predpoved
     .point.vektor, envir = parent.frame())
  point.vektor.test.arima <- window(dataset$mcp, start=c(1+24*(
     den-1)), end=c(24*(den-1)+24*delka))
  point.vektor.test.arima <- as.numeric(point.vektor.test.arima)
  assign(paste("point.vektor.test.arima",sep=""), point.vektor.
     test.arima , envir = parent.frame())
  predpoved <- point.vektor.predpoved.arima
  test <- point.vektor.test.arima
  predikce.predpoved.test.arima <- data.frame(predpoved)
  predikce.predpoved.test.arima <- rbind(predikce.predpoved.test
     .arima, test)
  assign(paste("predikce.predpoved.test.arima",sep=""), predikce
     .predpoved.test.arima, envir = parent.frame())
}

#STATISTIKY PREDPOVEDI
{
  #Vykresleni grafu predpoved-skutecnost
  point.vektor.predpoved.arima<-unlist(point.vektor.predpoved.
     arima,recursive = TRUE, use.names = FALSE)
  plot(point.vektor.predpoved.arima, type="l", col="red", main="
     Predikce a skutecnost MCP, den 301. ", ylim=c(-20,60), xlab
     ="hodina", ylab="MCP")
  lines(point.vektor.test.arima, type="l", col="black")

  #RMSE
  print(paste("RMSE: ", pocitej.rmse(point.vektor.predpoved,
     point.vektor.test), sep=""))

  #MAE
  print(paste("MAE: ", pocitej.mae(point.vektor.predpoved, point
     .vektor.test), sep=""))

  #MAPE
  print(paste("MAPE: ", pocitej.mape(point.vektor.predpoved,
```

```
    point.vektor.test), sep=""))

}
```

Kód 8: Výzkum třídenních předpovědí včetně parametrů odhadnutých modelů

```
#ROZSIRENY TRIDENNI PRUZKUM - STATISTIKY

{
  for(i in 200:(365-2)) {
    predpovidej(i,3)
    point.vektor.predpoved<-unlist(point.vektor.predpoved.arima,
        recursive = TRUE, use.names = FALSE)
    rmse<-pocitej.rmse(point.vektor.predpoved, point.vektor.test
        )
    mape<-pocitej.mape(point.vektor.predpoved, point.vektor.test
        )
    mae<-pocitej.mae(point.vektor.predpoved, point.vektor.test)
    den<-i

    #Extrakce parametru modelu do globalnich promennych
    jmena.sloupcu<-c("1","2","3","4","5","6","7","8","9","10","
        11","12","13","14","15","16","17","18","19","20","21","22
        ","23","24")

    assign(paste("sigma2_den",i,sep = ""),as.numeric(parametry.
        modelu$sigma2))
    assign(paste("p_q_test_den",i,sep = ""),as.numeric(parametry
        .modelu$q.rezidua))
    assign(paste("rad_arima",i,sep = ""),parametry.modelu$rad.
        arima)

    #Vytvoreni dataframu predpovedi a parametru modelu
    options(stringsAsFactors = FALSE)
    if(i==200) {
      tridenni.pruzkum<-data.frame(den,rmse,mape,mae)
      tridenni.pruzkum.sigma2<-data.frame(t(get(paste("sigma2_
          den",i,sep = ""))))
      tridenni.pruzkum.q.test<-data.frame(t(get(paste("p_q_test_
          den",i,sep = ""))))
      tridenni.pruzkum.arima<-data.frame(t(get(paste("rad_arima"
          ,i,sep = ""))))
    }
    else {
```

```
        tridenni.pruzkum<-rbind(tridenni.pruzkum,c(den,rmse,mape,
            mae))
        tridenni.pruzkum.sigma2<-rbind(tridenni.pruzkum.sigma2,c(t
            (get(paste("sigma2_den",i,sep = "")))))
        tridenni.pruzkum.q.test<-rbind(tridenni.pruzkum.q.test,c(t
            (get(paste("p_q_test_den",i,sep = "")))))
        tridenni.pruzkum.arima<-rbind(tridenni.pruzkum.arima,c(t(
            get(paste("rad_arima",i,sep = "")))))
      }

      colnames(tridenni.pruzkum.sigma2)<-jmena.sloupcu
      colnames(tridenni.pruzkum.q.test)<-jmena.sloupcu
      colnames(tridenni.pruzkum.arima)<-jmena.sloupcu

      print(paste("Prave spocitany den: ", i, sep=""))
    }
    write.csv2(tridenni.pruzkum, file="tridenni_pruzkum.csv")
    write.csv2(tridenni.pruzkum.sigma2, file="tridenni_pruzkum_
        sigma2.csv")
    write.csv2(tridenni.pruzkum.q.test, file="tridenni_pruzkum_q_
        test.csv")
    write.csv2(tridenni.pruzkum.arima, file="tridenni_pruzkum_
        arima.csv")
}
```

Kód 9: Analýza výzkumu třídenních předpovědí, graf

```
# ANALYZA TRIDENNIHO VYZKUMU

# Nahrani souboru
analyza.tridenni.rmse <- read.csv2(file=file.path("tridenni_
    pruzkum.csv"))

# Graf RMSE pro tridenni vyzkum
png(filename="analyza_200_363_RMSE(den).png")
plot(x<-analyza.tridenni.rmse$den, y<-analyza.tridenni.rmse$rmse
    , type="l", col="black", main = "RMSE tridenni predpovedi pro
     200. az 363. den roku", ylim=c(0,45), xlab="Pocatecni den
    tridenni predpovedi", ylab="RMSE")
dev.off()

# Prurezove statistiky RMSE
max(analyza.tridenni.rmse$rmse)
min(analyza.tridenni.rmse$rmse)
```

```
mean(analyza.tridenni.rmse$rmse)
var(analyza.tridenni.rmse$rmse)

# Nejmensi a nejvetsi RMSE pruzkumu
analyza.tridenni.rmse$den[which(analyza.tridenni.rmse$rmse==min(
    analyza.tridenni.rmse$rmse))]
# Min RMSE den 265
analyza.tridenni.rmse$den[which(analyza.tridenni.rmse$rmse==max(
    analyza.tridenni.rmse$rmse))]
# Min RMSE den 300

# Prurezove statistiky rozptylu pro jednotlive hodiny
analyza.tridenni.sigma2 <- read.csv2(file=file.path("tridenni_
    pruzkum_sigma2.csv"))
for(i in 1:24) {
  mean <- mean(analyza.tridenni.sigma2[,i+1])
  se <- sqrt(var(analyza.tridenni.sigma2[,i+1]))
  if(i==1) {
    tridenni.se <- data.frame(c(mean, se))
  }
  else {
    tridenni.se <- cbind(tridenni.se, c(mean, se))
  }
}
names(tridenni.se)<-c(NULL,"1","2","3","4","5","6","7","8","9","
    10","11","12","13","14","15","16","17","18","19","20","21","
    22","23","24")
write.csv2(tridenni.se, file="tridenni_sigma2_mean_se.csv")
```

Kód 10: Předpovědní funkce rozšíření $ARIMA-G$

```
# ARCH-efekt test z knihovny aTSA
arch.test <- function(residuals,output = TRUE)
{
  res <- residuals
  res2 <- res^2
  n <- length(res)
  if (n < 2L)
    stop("not enough length of residuals")
  lag.max <- min(24,n)
  lag.seq <- seq(ifelse(lag.max > 4,4,2),lag.max, by = 4)
  PQ <- LM <- p.value1 <- p.value2 <- numeric(length(lag.seq))
  for (i in 1:length(lag.seq)) {
    k <- round(sqrt(length(residuals)),0)
```

```
    LB.test <- Box.test(res2,lag = k)
    PQ[i] <- LB.test$statistic
    p.value1[i] <- LB.test$p.value
    rt <- embed(res2,k)
    lm.fit <- lm(rt[,1] ~ rt[,-1])
    SSE <- sum((rt[,1] - residuals(lm.fit))^2)
    SST <- sum((rt[,1] - mean(rt[,1]))^2)
    LM[i] <- ((SST - SSE)/k)/(SSE/(n - 2*k - 1))
    p.value2[i] <- 1 - pchisq(LM[i],k - 1)
  }
  PQcheck <- matrix(c(lag.seq,PQ,p.value1),ncol = 3)
  LMcheck <- matrix(c(LM,p.value2),ncol = 2)
  result <- cbind(PQcheck,LMcheck)
  colnames(result) <- c("order","PQ","p.value","LM","p.value")
  if (output) {
    cat("ARCH heteroscedasticity test for residuals","\n")
    cat("alternative: heteroscedastic","\n\n")
    cat("Portmanteau-Q test:","\n")
    print(result[,1:3],digits = 3)
    cat("Lagrange-Multiplier test:","\n")
    print(result[,c(1,4,5)],digits = 3)
    op <- par(mfrow = c(2,2))
    plot(1:n,res,xlab = "Time",ylab = "resid")
    abline(h = 0,col = 2,lty = 2)
    plot(1:n,res2,xlab = "Time",ylab = "resid^2")
    abline(h = 0,col = 2,lty = 2)
    plot(lag.seq,p.value1,ylim = c(0,1),xlab = "Order",ylab = "
      PQ prob.")
    abline(h = .05,col = 3, lty = 2)
    plot(lag.seq,p.value2,ylim = c(0,1),xlab = "Order",ylab = "
      LM prob.")
    abline(h = .05,col = 3, lty = 2)
    par(op)
  }
  arch.test <- result
}

#Predpovedni funkce ARIMA-G kompatibilni s vykreslovaci funkci
predpovidej.garch.plot <- function (den,delka) {
  settings<-read.csv2(file=file.path("source","settings.csv"))
  for(h in 1:24) {
    if(h < 10) {
      h.cislo.hodiny <- paste("h0", h, sep = "")
```

```
  jmeno.predpovedi <- paste("f0", h, sep = "")
}
else{
  h.cislo.hodiny <- paste("h", h, sep = "")
  jmeno.predpovedi <- paste("f", h, sep = "")
}


print(paste("### Hodina ", h, " ###", sep = ""))

#Jestli je chyba kpss pro jeho nejlepsi ucebni delku mensi
    nez u adf, predpovidej kpss
#Nulova hypoteza lm i portmanteau -
#Jako "portmanteau" je zde minen McLeod-Li test - v tomto
    kontextu ekvivalentni vyjadreni (portmanteu je vseobecne
    Q-test, McLeod-Li test take)

#Predpovedni algoritmus zcasti totozny z predchozim,
    zakomentovany rozdilnosti
if(settings$min.mean.hodnota.kpss[h] <= settings$min.mean.
    hodnota.adf[h]) {
  for.train=window(get(h.cislo.hodiny),start=c(den-settings$
      min.mean.rmse.kpss[h]),end=c(den-1))
  for.model=auto.arima(for.train, trace=FALSE, test="kpss",
      ic="aic", stepwise = FALSE, approximation = FALSE)

  # U odhadnuteho ARIMA modelu je testovana pritomnost
      heteroskedasticity
  arch<-arch.test(for.model$residuals, output = FALSE)
  arch.portmanteau.p<-unname(arch[1,3])
  arch.lm.p<-unname(arch[1,5])
  print(c("arch.portmanteau.p",arch.portmanteau.p))
  print(c("arch.lm.p",arch.lm.p))

  #Je-li detekovan ARCH efekt
  if(arch.portmanteau.p <= 0.05 || arch.lm.p <= 0.05) {
    #Je nejprve nastavena dostatecna delka uceni
    if(length(for.train) < 102) {
      for.train.garch = window(get(h.cislo.hodiny),start=c(
          den-102),end=c(den-1))
    }

    else {
```

```
  for.train.garch = for.train
}

# Spec funkci rugarch jsou predany parametry modelu z
    puvodni ARIMY
print(arimaorder(for.model))
print(for.rad.p<-arimaorder(for.model)[1])
print(for.rad.q<-arimaorder(for.model)[3])
print(for.rad.i<-arimaorder(for.model)[2])
garchspec <- ugarchspec(variance.model = list(model = "
    sGARCH", garchOrder = c(1,1)),
                            mean.model = list(armaOrder = c(
                                for.rad.p, for.rad.q)))

# Algoritmus ARIMA-G, diferencni vetveni
if(for.rad.i == 0) {
  print("2. KPSS, i=0")

  #V pripade, ze se nepovede nafitovat predpoved,
     pokracuje algoritmus jako puvodni predpovedni
     postup
  pokus<-tryCatch(
    {
      print(for.train.garch)
      #Nafitovani modelu dle specifik na ucici obdobi
      garchfit <- ugarchfit(spec = garchspec, data=for.
         train.garch)
      #Ucineni predpovedi s nafitovanym modelem
      garchforecast<-ugarchforecast(garchfit, n.ahead=
         delka)
      #Extrakce hodnot predpovedi a vytvoreni korektni
         casove rady z techto hodnot
      garch<-unname(garchforecast@forecast$seriesFor
         [,1])
      garch.ts<-ts(garch, start = c(den), end = c(den+
         delka-1))
      for.forecast<-garch.ts
      print(garchforecast@forecast$seriesFor)

      #Overeni rezidui Q-testem a McLeod-Li testem
      rezidua.q.p<-q.test.cipra(garchfit@fit$residuals)$
         p.value
      rezidua.q2.p<-q.test.cipra(garchfit@fit$residuals
```

```
            ^2)$p.value

          print(c("p hod Q-test",q.test.cipra(garchfit@fit$
              residuals)$p.value))
          print(c("p hod Q-test ^2 (portmanteau)",q.test.
              cipra(garchfit@fit$residuals^2)$p.value))

        },
        #Osetreni chyb
        error=function(cond) {
          message(cond)
          return(NA)
        },
        warning=function(cond) {
          message(cond)
          return(NULL)
        },
        finally = {
        }
      )
      #Pripad chyby fitovani ci selhani knihovny rugarch -
          predpoved dle auto.arima
      if(!is.null(pokus) && length(pokus) > 0 && is.na(pokus
          )) {
        for.forecast.list=forecast(for.model, h=delka)
        for.forecast<-for.forecast.list$mean
        rezidua.q.p<-NA
        rezidua.q2.p<-NA
      }
      #Pripad chyby fitovani ci selhani knihovny rugarch -
          predpoved dle auto.arima
      if(is.null(pokus)) {
        for.forecast.list=forecast(for.model, h=delka)
        for.forecast<-for.forecast.list$mean
        rezidua.q.p<-NA
        rezidua.q2.p<-NA
      }
    }

    if(for.rad.i == 1) {
      print("2. KPSS, i=1")
      pokus<-tryCatch(
        {
```

```
    for.t<-for.train.garch
    for.t.d1<-diff(for.t)
    print(for.t.d1)
    garchfit <- ugarchfit(spec = garchspec, data=for.t
        .d1)
    garchforecast<-ugarchforecast(garchfit, n.ahead=
        delka)
    garch<-unname(garchforecast@forecast$seriesFor
        [,1])
    garch.ts<-ts(garch, start = c(den), end = c(den+
        delka-1))

    obnoveno<-diffinv(garch.ts)
    obnoveno.c<-obnoveno+for.t[length(for.t)]
    obnoveno.c.for<-window(obnoveno.c, start=c(time(
        obnoveno.c)[2]), end=c(time(obnoveno.c)[length(
        time(obnoveno.c))]))
    for.forecast<-obnoveno.c.for

    rezidua.q.p<-q.test.cipra(garchfit@fit$residuals)$
        p.value
    rezidua.q2.p<-q.test.cipra(garchfit@fit$residuals
        ^2)$p.value

    print(c("p hod Q-test",q.test.cipra(garchfit@fit$
        residuals)$p.value))
    print(c("p hod Q-test ^2 (portmanteau)",q.test.
        cipra(garchfit@fit$residuals^2)$p.value))
  },
  error=function(cond) {
    message(cond)
    return(NA)
  },
  warning=function(cond) {
    message(cond)
    return(NULL)
  },
  finally = {
  }
)
if(!is.null(pokus) && length(pokus) > 0 && is.na(pokus
    )) {
  for.forecast.list=forecast(for.model, h=delka)
```

```
    for.forecast<-for.forecast.list$mean
    rezidua.q.p<-NA
    rezidua.q2.p<-NA
  }
  if(is.null(pokus)) {
    for.forecast.list=forecast(for.model, h=delka)
    for.forecast<-for.forecast.list$mean
    rezidua.q.p<-NA
    rezidua.q2.p<-NA
  }


}

if(for.rad.i == 2) {
  print("2. KPSS, i=2")
  pokus<-tryCatch(
    {
      for.t<-for.train.garch
      for.t.d1<-diff(for.t)
      for.t.d2<-diff(for.t.d1)
      print(for.t.d2)

      garchfit <- ugarchfit(spec = garchspec, data=for.t
          .d2)
      garchforecast<-ugarchforecast(garchfit, n.ahead=
          delka)
      garch<-unname(garchforecast@forecast$seriesFor
          [,1])
      garch.ts<-ts(garch, start = c(den), end = c(den+
          delka-1))

      diff.1<-diffinv(garch.ts)
      obnoveno.diff.c<-diff.1+for.t.d1[length(for.t.d1)]
      obnoveno.diff<-window(obnoveno.diff.c, start=c(
          time(obnoveno.diff.c)[2]), end=c(time(obnoveno.
          diff.c)[length(time(obnoveno.diff.c))]))

      obnoveno<-diffinv(obnoveno.diff)
      obnoveno.c<-obnoveno+for.t[length(for.t)]
      obnoveno.c.for<-window(obnoveno.c, start=c(time(
          obnoveno.c)[2]), end=c(time(obnoveno.c)[length(
          time(obnoveno.c))]))
```

```
        for.forecast<-obnoveno.c.for

        rezidua.q.p<-q.test.cipra(garchfit@fit$residuals)$
            p.value
        rezidua.q2.p<-q.test.cipra(garchfit@fit$residuals
            ^2)$p.value

        print(c("p hod Q-test",q.test.cipra(garchfit@fit$
            residuals)$p.value))
        print(c("p hod Q-test ^2 (portmanteau)",q.test.
            cipra(garchfit@fit$residuals^2)$p.value))
      },
      error=function(cond) {
        message(cond)
        return(NA)
      },
      warning=function(cond) {
        message(cond)
        return(NULL)
      },
      finally = {
      }
    )
    if(!is.null(pokus) && length(pokus) > 0 && is.na(pokus
        )) {
      for.forecast.list=forecast(for.model, h=delka)
      for.forecast<-for.forecast.list$mean
      rezidua.q.p<-NA
      rezidua.q2.p<-NA
    }

    if(is.null(pokus)) {
      for.forecast.list=forecast(for.model, h=delka)
      for.forecast<-for.forecast.list$mean
      rezidua.q.p<-NA
      rezidua.q2.p<-NA
    }

  }


}
```

```
  else {
    print("2. KPSS, no garch")
    for.forecast.list=forecast(for.model, h=delka)
    for.forecast<-for.forecast.list$mean

  }

  print(paste("h ", h, " kpss", sep = ""))
}
else {
  for.train=window(get(h.cislo.hodiny),start=c(den-settings$
     min.mean.rmse.adf[h]),end=c(den-1))
  for.model=auto.arima(for.train, trace=FALSE, test="adf",
     ic="aic", stepwise = FALSE, approximation = FALSE)
  arch<-arch.test(for.model$residuals, output = FALSE)
  arch.portmanteau.p<-unname(arch[1,3])
  arch.lm.p<-unname(arch[1,5])
  print(c("arch.portmanteau.p",arch.portmanteau.p))
  print(c("arch.lm.p",arch.lm.p))

  if(arch.portmanteau.p <= 0.05 || arch.lm.p <= 0.05) {
    if(length(for.train) < 102) {
      for.train.garch = window(get(h.cislo.hodiny),start=c(
         den-102),end=c(den-1))
    }
    else {
      for.train.garch = for.train
    }

    for.rad.p<-arimaorder(for.model)[1]
    for.rad.q<-arimaorder(for.model)[3]
    for.rad.i<-arimaorder(for.model)[2]
    garchspec <- ugarchspec(variance.model = list(model = "
       sGARCH", garchOrder = c(1,1)),
                            mean.model = list(armaOrder = c(
                               for.rad.p,for.rad.q)))

    if(for.rad.i == 0) {
      print("2.ADF i=0")
      pokus<-tryCatch(
        {
          garchfit <- ugarchfit(spec = garchspec, data=for.
             train.garch)
```

```
      garchforecast<-ugarchforecast(garchfit, n.ahead=
          delka)
      garch<-unname(garchforecast@forecast$seriesFor
          [,1])
      garch.ts<-ts(garch, start = c(den), end = c(den+
          delka-1))
      for.forecast<-garch.ts
      print(garchforecast@forecast$seriesFor)

      rezidua.q.p<-q.test.cipra(garchfit@fit$residuals)$
          p.value
      rezidua.q2.p<-q.test.cipra(garchfit@fit$residuals
          ^2)$p.value

      print(c("p hod Q-test",q.test.cipra(garchfit@fit$
          residuals)$p.value))
      print(c("p hod Q-test ^2 (portmanteau)",q.test.
          cipra(garchfit@fit$residuals^2)$p.value))
    },
    error=function(cond) {
      message(cond)
      return(NA)
    },
    warning=function(cond) {
      message(cond)
      return(NULL)
    },
    finally = {
    }
)
if(!is.null(pokus) && length(pokus) > 0 && is.na(pokus
    )) {
    for.forecast.list=forecast(for.model, h=delka)
    for.forecast<-for.forecast.list$mean
    rezidua.q.p<-NA
    rezidua.q2.p<-NA
}

if(is.null(pokus)) {
    for.forecast.list=forecast(for.model, h=delka)
    for.forecast<-for.forecast.list$mean
    rezidua.q.p<-NA
    rezidua.q2.p<-NA
```

```
  }
}

if(for.rad.i == 1) {
  print("2.ADF i=1")
  pokus<-tryCatch(
    {
      assign(paste("for.t.d1.test",sep=""), for.t.d1,
          envir = parent.frame())
      for.t<-for.train.garch
      for.t.d1<-diff(for.t)

      garchfit <- ugarchfit(spec = garchspec, data=for.t
          .d1)
      garchforecast<-ugarchforecast(garchfit, n.ahead=
          delka)
      garch<-unname(garchforecast@forecast$seriesFor
          [,1])
      garch.ts<-ts(garch, start = c(den), end = c(den+
          delka-1))

      obnoveno<-diffinv(garch.ts)
      obnoveno.c<-obnoveno+for.t[length(for.t)]
      obnoveno.c.for<-window(obnoveno.c, start=c(time(
          obnoveno.c)[2]), end=c(time(obnoveno.c)[length(
          time(obnoveno.c))]))
      for.forecast<-obnoveno.c.for

      rezidua.q.p<-q.test.cipra(garchfit@fit$residuals)$
          p.value
      rezidua.q2.p<-q.test.cipra(garchfit@fit$residuals
          ^2)$p.value

      print(c("p hod Q-test",q.test.cipra(garchfit@fit$
          residuals)$p.value))
      print(c("p hod Q-test ^2 (portmanteau)",q.test.
          cipra(garchfit@fit$residuals^2)$p.value))
    },
    error=function(cond) {
      message(cond)
      return(NA)
    },
    warning=function(cond) {
```

```
      message(cond)
      return(NULL)
    },
    finally = {
    }
  )
  if(!is.null(pokus) && length(pokus) > 0 && is.na(pokus
    )) {
    for.forecast.list=forecast(for.model, h=delka)
    for.forecast<-for.forecast.list$mean
    rezidua.q.p<-NA
    rezidua.q2.p<-NA
  }

  if(is.null(pokus)) {
    for.forecast.list=forecast(for.model, h=delka)
    for.forecast<-for.forecast.list$mean
    rezidua.q.p<-NA
    rezidua.q2.p<-NA
  }
}

if(for.rad.i == 2) {
  print("2.ADF i=2")
  pokus<-tryCatch(
    {
      for.t<-for.train.garch
      for.t.d1<-diff(for.t)
      for.t.d2<-diff(for.t.d1)

      garchfit <- ugarchfit(spec = garchspec, data=for.t
          .d2)
      garchforecast<-ugarchforecast(garchfit, n.ahead=
          delka)
      garch<-unname(garchforecast@forecast$seriesFor
          [,1])
      garch.ts<-ts(garch, start = c(den), end = c(den+
          delka-1))

      diff.1<-diffinv(garch.ts)
      obnoveno.diff.c<-diff.1+for.t.d1[length(for.t.d1)]
      obnoveno.diff<-window(obnoveno.diff.c, start=c(
          time(obnoveno.diff.c)[2]), end=c(time(obnoveno.
```

```
        diff.c)[length(time(obnoveno.diff.c))]))

    obnoveno<-diffinv(obnoveno.diff)
    obnoveno.c<-obnoveno+for.t[length(for.t)]
    obnoveno.c.for<-window(obnoveno.c, start=c(time(
        obnoveno.c)[2]), end=c(time(obnoveno.c)[length(
        time(obnoveno.c))]))
    for.forecast<-obnoveno.c.for

    rezidua.q.p<-q.test.cipra(garchfit@fit$residuals)$
        p.value
    rezidua.q2.p<-q.test.cipra(garchfit@fit$residuals
        ^2)$p.value

    print(c("p hod Q-test",q.test.cipra(garchfit@fit$
        residuals)$p.value))
    print(c("p hod Q-test ^2 (portmanteau)",q.test.
        cipra(garchfit@fit$residuals^2)$p.value))
  },
  error=function(cond) {
    message(cond)
    return(NA)
  },
  warning=function(cond) {
    message(cond)
    return(NULL)
  },
  finally = {
  }
)
if(!is.null(pokus) && length(pokus) > 0 && is.na(pokus
    )) {
  for.forecast.list=forecast(for.model, h=delka)
  for.forecast<-for.forecast.list$mean
  rezidua.q.p<-NA
  rezidua.q2.p<-NA
}

if(is.null(pokus)) {
  for.forecast.list=forecast(for.model, h=delka)
  for.forecast<-for.forecast.list$mean
  rezidua.q.p<-NA
  rezidua.q2.p<-NA
```

```
      }
    }


  }

  else {
    print("2.ADF, no garch")
    for.forecast.list=forecast(for.model, h=delka)
    for.forecast<-for.forecast.list$mean
  }

  print(paste("h ", h, " adf", sep = ""))
}


assign(jmeno.predpovedi, for.forecast, envir = parent.frame
    ())
print(jmeno.predpovedi)
print(for.forecast)

#Extrakce parametru modelu

koeficienty<-for.model$coef
koeficienty_jmena<-names(koeficienty)
koeficienty_cisla<-unname(koeficienty)

sigma2<-round(for.model$sigma2,3)
q.rezidua<-round(q.test.cipra(for.model$residuals)$p.value
    ,3)
rad.arima<-paste("(",toString(arimaorder(for.model)),")",sep
    = "")
koef.jmena<-toString(koeficienty_jmena)
koef.cisla<-toString(round(koeficienty_cisla,3))

options(stringsAsFactors = FALSE)

if (h==1) {
  parametry.modelu.arima=data.frame(h,sigma2,q.rezidua,rad.
      arima,koef.jmena,koef.cisla)
  parametry.modelu.garch=data.frame(h,arch.lm.p,arch.
      portmanteau.p,rezidua.q.p,rezidua.q2.p)
```

```
  }
  else {
    parametry.modelu.arima<-rbind(parametry.modelu.arima,c(h,
        sigma2,q.rezidua,rad.arima,koef.jmena,koef.cisla))
    parametry.modelu.garch<-rbind(parametry.modelu.garch,c(h,
        arch.lm.p,arch.portmanteau.p,rezidua.q.p,rezidua.q2.p))
  }

}

assign(paste("parametry.modelu.arima",sep=""), parametry.
    modelu.arima, envir = parent.frame())
assign(paste("parametry.modelu.garch",sep=""), parametry.
    modelu.garch, envir = parent.frame())


#Vytvoreni predpovedni tabulky

forecast.table.method<-data.frame(f01,f02,f03,f04,f05,f06,f07,
    f08,f09,f10,f11,
                                  f12,f13,f14,f15,f16,f17,f18,
                                      f19,f20,f21,f22,f23,f24)

assign(paste("forecast.table",sep=""), forecast.table.method,
    envir = parent.frame())

for (i in 1:delka) {
  if(i == 1) {
    predpoved.point.vektor <- forecast.table[i,]
  }
  else {
    predpoved.point.vektor <- c(predpoved.point.vektor,
        forecast.table[i,])
  }
}

#Tisk a zapsani ziskanych predpovedi do globalnich promennych
print(predpoved.point.vektor)
assign(paste("point.vektor.predpoved.garch",sep=""), predpoved
    .point.vektor, envir = parent.frame())
point.vektor.test.garch<-window(dataset$mcp, start=c(1+24*(den
    -1)), end=c(24*(den-1)+24*delka))
point.vektor.test.garch<-as.numeric(point.vektor.test.garch)
```

```
  assign(paste("point.vektor.test.garch",sep=""), point.vektor.
     test.garch , envir = parent.frame())
  predpoved <- point.vektor.predpoved.garch
  test <- point.vektor.test.garch
  predikce.predpoved.test.garch <- data.frame(predpoved)
  predikce.predpoved.test.garch <- rbind(predikce.predpoved.test
     .garch, test)
  assign(paste("predikce.predpoved.test.garch",sep=""), predikce
     .predpoved.test.garch, envir = parent.frame())
}

#STATISTIKY PREDPOVEDI ARIMA-GARCH
{
  point.vektor.predpoved.arima<-unlist(point.vektor.predpoved.
     arima,recursive = TRUE, use.names = FALSE)
  point.vektor.predpoved.garch<-unlist(point.vektor.predpoved.
     garch,recursive = TRUE, use.names = FALSE)
  plot(point.vektor.predpoved.garch, type="l", col="red", main="
     Predikce a skutecnost MCP, den 332. - 334.", ylim=c(0,120),
      xlab="hodina", ylab="MCP")
  lines(point.vektor.predpoved.arima, type="l", col="dodgerblue"
     )
  lines(point.vektor.test.arima, type="l", col="black")
  lines(point.vektor.test.garch, type="l", col="black")

  print(c("Test - shodna pozice predpovedi ", min(point.vektor.
     test.arima-point.vektor.test.garch), max(point.vektor.test.
     arima-point.vektor.test.garch)))
  print(c("RMSE ARIMA ", round(pocitej.rmse(point.vektor.
     predpoved.arima,point.vektor.test.arima),2)))
  print(c("RMSE GARCH ", round(pocitej.rmse(point.vektor.
     predpoved.garch,point.vektor.test.garch),2)))
  print(c("RMSE GARCH-ARIMA ", pocitej.rmse(point.vektor.
     predpoved.garch,point.vektor.test.garch) - pocitej.rmse(
     point.vektor.predpoved.arima,point.vektor.test.arima)))
  print(c("RMSE GARCH-ARIMA ", round(pocitej.rmse(point.vektor.
     predpoved.garch,point.vektor.test.garch) - pocitej.rmse(
     point.vektor.predpoved.arima,point.vektor.test.arima),2)))

  arima.garch.test.export=rbind(point.vektor.predpoved.arima,
     point.vektor.predpoved.garch,point.vektor.test.arima)
}
```

Kód 11: Analýza výzkumu třídenních předpovědí ARIMA a ARIMA-G, graf

```
#Realizace a ulozeni tridenniho vyzkumu RMSE GARCH
{
  for(i in 200:(363)) {
    predpovidej.garch.plot(i,3)
    point.vektor.predpoved.garch<-unlist(point.vektor.predpoved.
       garch,recursive = TRUE, use.names = FALSE)
    rmse<-pocitej.rmse(point.vektor.predpoved.garch, point.
       vektor.test.garch)
    den<-i
    print(c("RMSE",rmse))

    if(i==200) {
      tridenni.pruzkum.garch<-data.frame(den,rmse)
    }
    else {
      tridenni.pruzkum.garch<-rbind(tridenni.pruzkum.garch,c(den
         ,rmse))
    }

    print(paste("Prave spocitany den: ", i, sep=""))
  }
  write.csv2(tridenni.pruzkum.garch, file="tridenni_pruzkum_
     garch(1,1)_200_363.csv")
}

#Analyza ARIMA-G v souvislosti s ARIMA
tridenni.garch<-read.csv2(file=file.path("tridenni_pruzkum_garch
   (1,1)_200_363.csv"))
tridenni.arima <- read.csv2(file=file.path("tridenni_pruzkum.csv
   "))

#Prurezove statistiky pro nerozsirene modely ARIMA
max(tridenni.arima$rmse)
min(tridenni.arima$rmse)
mean(tridenni.arima$rmse)
var(tridenni.arima$rmse)

#Vytvoreni caseove rady rozdilu chyb ARIMA vs. ARIMA-G
rozdil.rmse<-tridenni.garch$rmse-tridenni.arima$rmse

#Nalezeni dnu s nejvetsim zlepsenim a zhorsenim diky ARIMA-G
```

```
tridenni.garch$den[which(tridenni.garch$rmse==max(tridenni.garch
    $rmse))]
tridenni.garch$den[which(tridenni.garch$rmse==min(tridenni.garch
    $rmse))]

#Prurezove statistiky zmen chyb pri rozsireni o funkcinalitu
    GARCH
round(max(rozdil.rmse),2)
round(min(rozdil.rmse),2)
round(mean(rozdil.rmse),2)
round(sd(rozdil.rmse),2)

#Zlepseni zhorseni z poctu pripadu 164
#ARIMA lepsi 94
sum(rozdil.rmse > 0)
#GARCH lepsi 70
sum(rozdil.rmse < 0)
#Nedoslo ke zmene 0
sum(rozdil.rmse = 0)

#Prurezove statistiky chybovosti obou algorimu
round(max(tridenni.garch$rmse),2)
round(min(tridenni.garch$rmse),2)
round(mean(tridenni.garch$rmse),2)
round(sd(tridenni.garch$rmse),2)

round(max(tridenni.arima$rmse),2)
round(min(tridenni.arima$rmse),2)
round(mean(tridenni.arima$rmse),2)
round(sd(tridenni.arima$rmse),2)

#Nalezeni druheho nejvetsiho zhorseni GARCH
n <- length(rozdil.rmse)
rmse.descending <- sort(rozdil.rmse, decreasing = TRUE)
tridenni.arima$den[which(rozdil.rmse==rmse.descending[2])]

#Den s nejvetsim zlepsenim GARCH, overeni v obou casovych radach
    - stejny den 325
tridenni.arima$den[which(rozdil.rmse==min(rozdil.rmse))]
tridenni.garch$den[which(rozdil.rmse==min(rozdil.rmse))]

#Export chyb pro tento den
round(tridenni.arima$rmse[which(rozdil.rmse==min(rozdil.rmse))
```

```
    ],2)
round(tridenni.garch$rmse[which(rozdil.rmse==min(rozdil.rmse))
    ],2)

#kontrola nejlepsi GARCH - v abs() musi se rovnat maximu v rade
    rozdilu GARCH
round(tridenni.arima$rmse[which(rozdil.rmse==min(rozdil.rmse))]-
    tridenni.garch$rmse[which(rozdil.rmse==min(rozdil.rmse))],2)

#Den s nejvetsim zhorseni GARCH, overeni v obou casovych radach
    - stejny den 312
tridenni.arima$den[which(rozdil.rmse==max(rozdil.rmse))]
tridenni.garch$den[which(rozdil.rmse==max(rozdil.rmse))]

#Export chyb pro tento den
round(tridenni.arima$rmse[which(rozdil.rmse==max(rozdil.rmse))
    ],2)
round(tridenni.garch$rmse[which(rozdil.rmse==max(rozdil.rmse))
    ],2)

#kontrola nejhorsi GARCH
round(tridenni.arima$rmse[which(rozdil.rmse==max(rozdil.rmse))]-
    tridenni.garch$rmse[which(rozdil.rmse==max(rozdil.rmse))],2)

#Pozn. Export chyb pro den 332. proveden z vyhodnovaci a
    vykreslovaci funkce

#Export grafu vyzkumu chyb RMSE tridennich predpovedi pro ARIMA
    a ARIMA-G
png(filename="analyza_200_363_RMSE_ARIMA_GARCH.png")
plot(x<-tridenni.arima$den, y<-tridenni.arima$rmse, type="l",
    col="dodgerblue", main = "RMSE tridenni predpovedi pro 200.
    az 363. den roku", ylim=c(0,45), xlab="Pocatecni den tridenni
     predpovedi", ylab="RMSE", cex.lab=1.3, cex.axis=1.3, cex.
    main=1.3, cex.sub=1.3)
lines(x<-tridenni.garch$den, y<-tridenni.garch$rmse, type="l",
    col="red")
dev.off()
```