HW6 q2

10/7/2019

a)

We are interested in predicting arrival delay (ARR_DELAY) using the variables that would be available before the flight takes off: departure time, departure airport (ORD or DFW), day of the month, and air carrier (airline). Also, make a "day of week" variable. First, for a baseline comparison, fit a linear model to the data using these variables. (Be naive: assume the effects of time of day or day of week are linear.) Show a table of the coefficients you find. Evaluate the model on the test set (using predict) and report its squared-error loss.

For efficicency I used only 1000 rows of the data. Coefficients are given in Table 1. We see that the mean squared error is 2503.734.

```
set.seed(1234)
rows=sample(nrow(train), 1000)
# fit on the training data
best.fit=lm(ARR_DELAY~ DEP_TIME +as.factor(ORIGIN) +
              DAY_OF_MONTH +as.factor(CARRIER) +
              as.factor(DAY_OF_WEEK), data=train[rows,])
#coef(best.fit, 8)
stargazer(best.fit,
          title = "Q2a",
          header=FALSE, type = "latex", font.size="small",
          column.sep.width = "1pt",
          single.row = TRUE)
```

```
# squared-error loss
mean((test$ARR_DELAY - predict.lm(best.fit, newdata = test))^2, na.rm = TRUE)
```

[1] 2896.995

b)

The relationships may be complicated and a linear model may not be appropriate, so use npreg to fit a Nadaraya–Watson kernel regression model; allow npregbw to select all bandwidths with cross-validation. Report the kernel regression's performance on the test set (again using predict and squared-error loss) and compare to the linear model. Does this method seem to do dra- matically better?

Coefficients are given in Table 2. We see that the mean squared error is 3309.925. This method does not seem to do better based on mse.

```
bw =npregbw(ARR_DELAY~ DEP_TIME +as.factor(ORIGIN) +
              DAY_OF_MONTH +as.factor(CARRIER) +
              as.factor(DAY_OF_WEEK),
            data=train[rows,])
```

Multistart 1 of 5 | Multistart 1 of 5 | Multistart 1 of 5 | Multistart 1 of 5 / Multistart 1 of 5 - Multistart 1 of 5
Multistart 1 of 5 | Multistart 1 of 5 / Multistart 1 of 5 - Multistart 1 of 5 | Multistart 1 of 5 | Multistart 1 of 5 / Multistart 1 of 5 - Multistart 2 of 5 | Multistart 2 of 5 | Multistart 2 of 5 / Multistart 2 of 5 - Multistart 2 of 5
Multistart 2 of 5 | Multistart 2 of 5 / Multistart 2 of 5 - Multistart 2 of 5 | Multistart 2 of 5 | Multistart 3 of 5 | Multistart 3 of 5 | Multistart 3 of 5 / Multistart 3 of 5 - Multistart 3 of 5

Table 1: Q2a

	Dependent variable:
	ARR_DELAY
DEP_TIME	0.009** (0.004)
as.factor(ORIGIN)ORD	5.243 (4.180)
DAY_OF_MONTH	0.042 (0.188)
as.factor(CARRIER)AS	23.595 (17.745)
as.factor(CARRIER)B6	−12.651 (20.214)
as.factor(CARRIER)DL	41.594*** (11.818)
as.factor(CARRIER)EV	2.474 (5.249)
as.factor(CARRIER)F9	−0.282 (18.921)
as.factor(CARRIER)NK	0.819 (8.709)
as.factor(CARRIER)OO	4.577 (6.425)
as.factor(CARRIER)UA	6.454 (5.294)
as.factor(CARRIER)VX	17.479 (30.648)
as.factor(DAY_OF_WEEK)1	2.196 (6.253)
as.factor(DAY_OF_WEEK)2	5.338 (6.470)
as.factor(DAY_OF_WEEK)3	9.279 (6.871)
as.factor(DAY_OF_WEEK)4	29.888*** (7.342)
as.factor(DAY_OF_WEEK)5	9.335 (6.770)
as.factor(DAY_OF_WEEK)6	0.167 (6.662)
Constant	−14.926* (8.346)
Observations	966
R^2	0.048
Adjusted R^2	0.029
Residual Std. Error	52.503 (df = 947)
F Statistic	2.628*** (df = 18; 947)
Note:	*p<0.1; **p<0.05; ***p<0.01

Multistart 3 of 5 | Multistart 3 of 5 / Multistart 3 of 5 - Multistart 3 of 5 | Multistart 3 of 5 | Multistart 4 of 5 | Multistart 4 of 5 | Multistart 4 of 5 / Multistart 4 of 5 - Multistart 4 of 5
Multistart 4 of 5 | Multistart 4 of 5 / Multistart 4 of 5 - Multistart 4 of 5 | Multistart 4 of 5 | Multistart 4 of 5 / Multistart 4 of 5 - Multistart 4 of 5
Multistart 4 of 5 | Multistart 4 of 5 / Multistart 5 of 5 | Multistart 5 of 5 | Multistart 5 of 5 / Multistart 5 of 5 - Multistart 5 of 5
Multistart 5 of 5 | Multistart 5 of 5 / Multistart 5 of 5 | Multistart 5 of 5 | Multistart 5 of 5 / Multistart 5 of 5 - Multistart 5 of 5
Multistart 5 of 5 | Multistart 5 of 5 /

```
loclinfit <- npreg(bw)
summary(loclinfit)
```

Regression Data: 966 training points, in 5 variable(s)

No. Complete Observations: 966 No. Incomplete (NA) Observations: 34 Observations omitted or excluded: 7 31 34 38 42 48 55 65 122 131 212 271 287 298 357 378 459 468 484 487 542 572 682 685 689 760 762 773 819 836 871 933 937 946 DEP_TIME as.factor(ORIGIN) DAY_OF_MONTH as.factor(CARRIER) Bandwidth(s): 55.30875 3.837364e-08 8.127438 0.6824608 as.factor(DAY_OF_WEEK) Bandwidth(s): 0.3799275

Kernel Regression Estimator: Local-Constant Bandwidth Type: Fixed Residual standard error: 22.66941 R-squared: 0.831108

Continuous Kernel Type: Second-Order Gaussian No. Continuous Explanatory Vars.: 2

Unordered Categorical Kernel Type: Aitchison and Aitken No. Unordered Categorical Explanatory Vars.: 3

```
# mean squared-error loss
mean((test$ARR_DELAY - predict(loclinfit, newdata = test))^2,
     na.rm = TRUE)
```

[1] 4269.089

c)

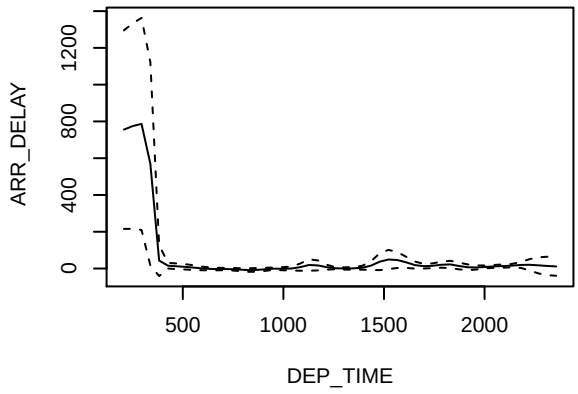
The plot function for npregression objects (such as the fit returned by npreg) can plot the marginal association of each variable with the response. If you set the plot.errors.method = "bootstrap" option, it will also plot bootstrap-based standard errors for these. Make the plots with standard errors and interpret the results. Which variables seem strongly related with delay length? What do the plots suggest about the appropriate-ness of linear regression? If you saw major non-linearities in any variable, do these non- linearities appear to harm the predictions enough to make linear regression perform dra- matically worse than kernel regression?

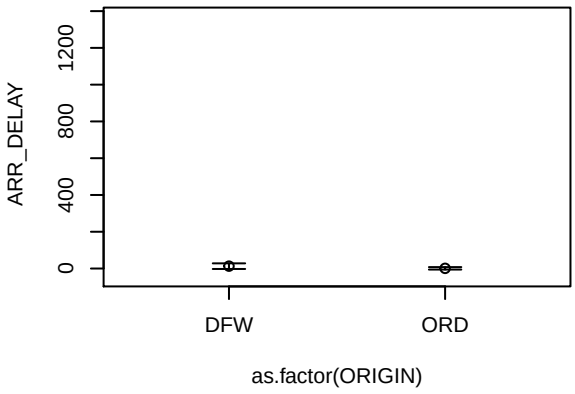
The plots below suggest that:

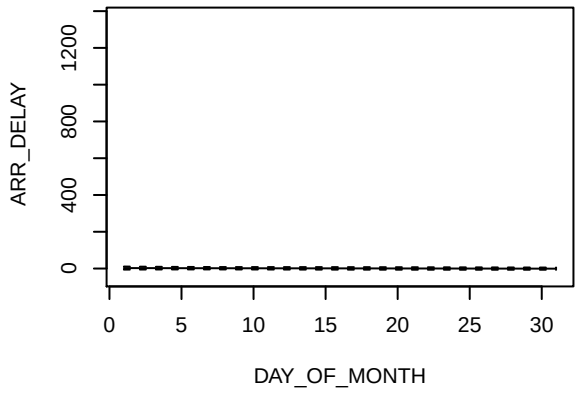
1. delay length (ARR_DELAY) is correlated with the variable ORIGIN. Seems like delay is longer from ORD.
2. delay length seems to decrease non linearly with DEP_TIME.

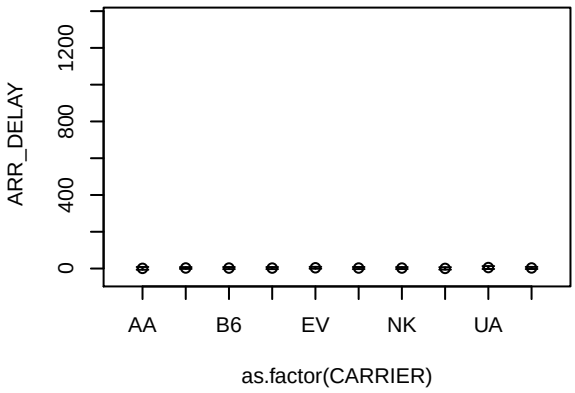
However the nonlinearities are slight and apparently don't seem to impair the performance of the ordinary linear regression for the subset of data we are using.

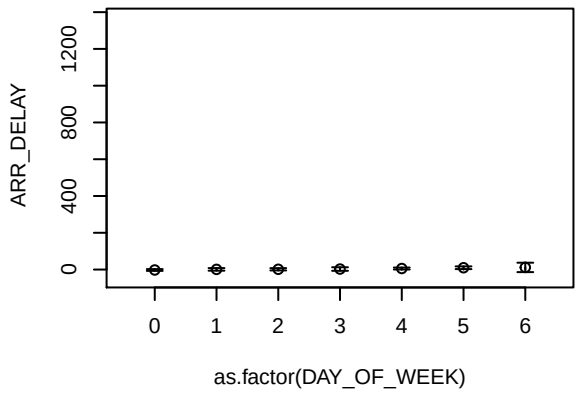
```
par(mfrow=c(3,2))
npplot(bw,
       data =data, plot.errors.method= "bootstrap")
```











d)

Repeat this analysis, but use a locally linear kernel regression (with regtype = "ll" provided to npregbw). Compare the test error from this model to that for the previous one. Discuss possible reasons for any difference you see.

Here it seems that the mean squared error is 3749.95 which is larger than the previous model. This suggests that the locally linear version doesn't perform as well as the local-constant kernel, maybe due to overfitting to the training data. We have also seen the linear model does not seem to perform worse using the training

data that we do have.

```
bw2=npregbw(ARR_DELAY~DEP_TIME +as.factor(ORIGIN) +
              DAY_OF_MONTH +as.factor(CARRIER) +
              as.factor(DAY_OF_WEEK),
            data=train[rows,],
            regtype = "ll")
```

Multistart 1 of 5 | Multistart 1 of 5 | Multistart 1 of 5 | Multistart 1 of 5 / Multistart 1 of 5 - Multistart 1 of 5
Multistart 1 of 5 | Multistart 1 of 5 / Multistart 1 of 5 | Multistart 1 of 5 | Multistart 1 of 5 / Multistart 1 of 5 - Multistart 1 of 5
Multistart 2 of 5 | Multistart 2 of 5 | Multistart 2 of 5 / Multistart 2 of 5 - Multistart 2 of 5
Multistart 2 of 5 | Multistart 2 of 5 / Multistart 2 of 5 - Multistart 2 of 5 | Multistart 2 of 5 | Multistart 2 of 5 / Multistart 2 of 5 - Multistart 2 of 5
Multistart 2 of 5 | Multistart 3 of 5 | Multistart 3 of 5 | Multistart 3 of 5 / Multistart 3 of 5 - Multistart 3 of 5
Multistart 3 of 5 | Multistart 3 of 5 / Multistart 3 of 5 | Multistart 3 of 5 | Multistart 3 of 5 / Multistart 3 of 5 - Multistart 3 of 5
Multistart 4 of 5 | Multistart 4 of 5 | Multistart 4 of 5 / Multistart 4 of 5 - Multistart 4 of 5
Multistart 4 of 5 | Multistart 4 of 5 / Multistart 4 of 5 | Multistart 4 of 5 | Multistart 4 of 5 / Multistart 4 of 5 - Multistart 4 of 5
Multistart 5 of 5 | Multistart 5 of 5 | Multistart 5 of 5 / Multistart 5 of 5 - Multistart 5 of 5
Multistart 5 of 5 | Multistart 5 of 5 / Multistart 5 of 5 | Multistart 5 of 5 | Multistart 5 of 5 / Multistart 5 of 5 - Multistart 5 of 5
Multistart 5 of 5 |

```
loclinfit <- npreg(bw2)
summary(loclinfit)
```

Regression Data: 966 training points, in 5 variable(s)

No. Complete Observations: 966 No. Incomplete (NA) Observations: 34 Observations omitted or excluded: 7 31 34 38 42 48 55 65 122 131 212 271 287 298 357 378 459 468 484 487 542 572 682 685 689 760 762 773 819 836 871 933 937 946 DEP_TIME as.factor(ORIGIN) DAY_OF_MONTH as.factor(CARRIER) Bandwidth(s): 68.88475 0.07024631 27.68321 0.6705578 as.factor(DAY_OF_WEEK) Bandwidth(s): 0.3985664

Kernel Regression Estimator: Local-Linear Bandwidth Type: Fixed Residual standard error: 24.57963 R-squared: 0.7962304

Continuous Kernel Type: Second-Order Gaussian No. Continuous Explanatory Vars.: 2

Unordered Categorical Kernel Type: Aitchison and Aitken No. Unordered Categorical Explanatory Vars.: 3

```
# squared-error loss
mean((test$ARR_DELAY - predict(loclinfit, newdata = test))^2, na.rm = TRUE)
```

[1] 30644.95

```
dev.new(width=5, height=8, unit = "in")
par(mfrow=c(5,1))
npplot(bw2,data = data,
       plot.errors.method= "bootstrap")
```