MATH 6359, Statistical Computing, Homework 5

Andrey Skripnikov

November 2, 2017

SUBMISSION GUIDELINES:

- **Bring the hard-copy of typed up solutions to the class on Tuesday, November 14.**
- **Keep it under 4 pages total (all included - text, code, plots, tables). DON'T (!) repeat the problem formulation, go straight to solution.**
- **Follow the example format in terms of conciseness.**
- **Point total is 45 (100%), and on top of that one can get 3 extra credit points total.**

PROBLEM #1 - 20 points (+3 EXTRA).

Find a data set containing a BINARY variable (will act as a RESPONSE) and multiple predictor variables (no matter if those are factors or continuous). It CAN'T be from $ISwR$ **package or from the lecture slides. For this data set:**

- **Write down the algebraic formula for logistic regression of your response variable on predictors, describe all the variables in the formula.**
- **Perform this logistic regression in R, report significant predictors.**
- **Interpret the TWO predictors that demonstrate LOWEST p-values (see example).**

Example: I will look at the *Titanic* data set, containing data on passenger survival (**binary response**) alongside with their age, gender and the class of their travel (all of those will act as our predictors).

Code:

```
> Titanic <- read.csv("/home/usdandres/Titanic.csv")
> Titanic.glm <- glm(Survived ~ PClass+Age+Sex,family=binomial,data=Titanic)
> summary(Titanic.glm)
Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  3.759662   0.397567   9.457  < 2e-16 ***
PClass2nd   -1.291962   0.260076  -4.968 6.78e-07 ***
PClass3rd   -2.521419   0.276657  -9.114  < 2e-16 ***
Age         -0.039177   0.007616  -5.144 2.69e-07 ***
Sexmale     -2.631357   0.201505 -13.058  < 2e-16 ***
```

Analysis:

1. **Response:** Let y_i denote whether the i^{th} passenger survived ($y_i = 1$) or not ($y_i = 0$).
2. **Predictors:** Let $(w_{1,i}, w_{2,i}, x_{1,i}, z_{1,i})$ denote the observed values of **dummy variables** $w_{j,i} = I(Class\ of\ i^{th}\ passenger\ = (j+1)), j = 1, 2$**, age** x_i **and gender** z_i for the i^{th} passenger.
3. **Probability:** Let $\pi_i = P(y_i = 1)$ - probability of survival for a passenger with predictor values $(w_{1,i}, w_{2,i}, x_{1,i}, z_{1,i})$.
4. **Logistic regression formula:**

$$logit(\pi_i) = \beta_0 + \beta_1 w_{1,i} + \beta_2 w_{2,i} + \beta_3 x_{1,i} + \beta_4 z_{1,i},\ \ i = 1, \ldots, n$$

5. Logistic regression revealed all the predictors - gender, age, class - to play a role in passenger's survival, due to their tiny p-values.
6. The smallest p-values correspond to β_2 ($PClass3rd$) and β_5 ($Sexmale$), let's interpret their coefficients:
   - $\beta_2 = -2.52, \implies$ **log-odds** of surviving are 2.52 less for third class passengers as opposed to first class
   - $\beta_5 = -2.63 \implies$ **log-odds** of surviving are less by 2.63 for males as opposed to females
7. **+3 BONUS POINTS** - also for those top-2 most significant variables, obtain the **odds ratios** instead of the **LOG-ODDS** ratios (as is originally outputted), and **interpret those odds ratios in plain English**

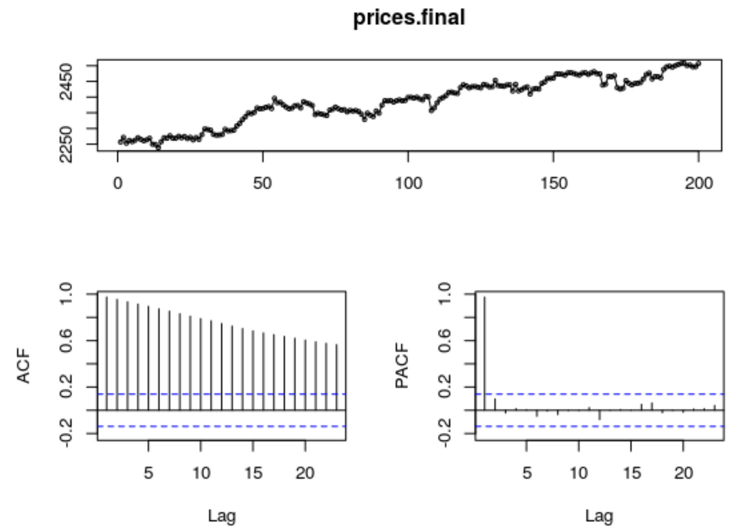
PROBLEM #2 - 25 points.

Find a univariate time series data set (shouldn't be an issue at all). For this data set:

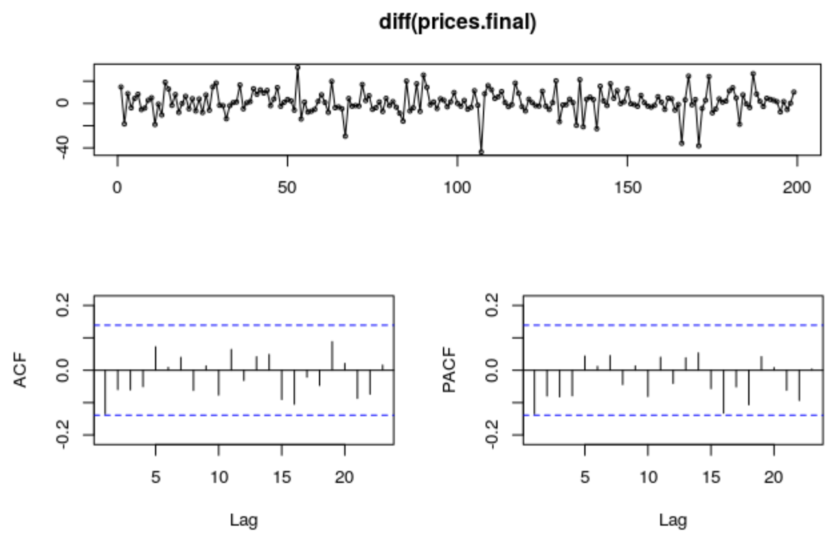
1. **Plot it and say if you witness variance-instability - clear changes in variability of observed values with time. If yes - apply log transform. If no - simply move on to the next step.**
2. **From that same plot - judge if there is seasonality. If yes, then verify it by looking at the ACF plot for this series - it will have spikes at seasonal lags (e.g. at lags 12, 24,... for $m = 12$ monthly data)**
3. **If there is seasonality with period m - apply seasonal difference with period m. NOTE - you DON'T HAVE to force yourself and look for time series with seasonality. If you don't detect seasonality - it is fine, just move on to the next steps.**
4. **Now, check if the (seasonally-differenced) series (denote it t_{sadj}) is stationary, by looking at the ACF plot. If not - apply first-order differencing ($d = 1$). Check the ACF plot of differenced time series, make sure it is stationary.**
5. **Once you get a stationary time series, look at its ACF and PACF plots, try to estimate the order p and q of your ARIMA model from those (see the slides for the guidelines of using ACF/PACF to judge p and a).**
6. **Having determined p and q, alongside d - number of times your applied first-order differences from step 4 - fit your $ARIMA(p, d, q)$ to the (seasonally-differenced) series t_{sadj} and plot the forecasts.**

Example: I will look at daily S&P 500 data over the last 200 days.
Code:

```
> library(forecast)
> library(quantmod)
> pr <- getSymbols("^GSPC",auto=F,from="2012-09-28", to="2017-09-28")[,4]
> prices <- as.numeric(pr); n.tp <- 1:length(prices)
> prices.final <- tail(prices,200)
> tsdisplay(prices.final)  # No seasonality or variance instability.
>                          # BUT clearly not stationary..
```

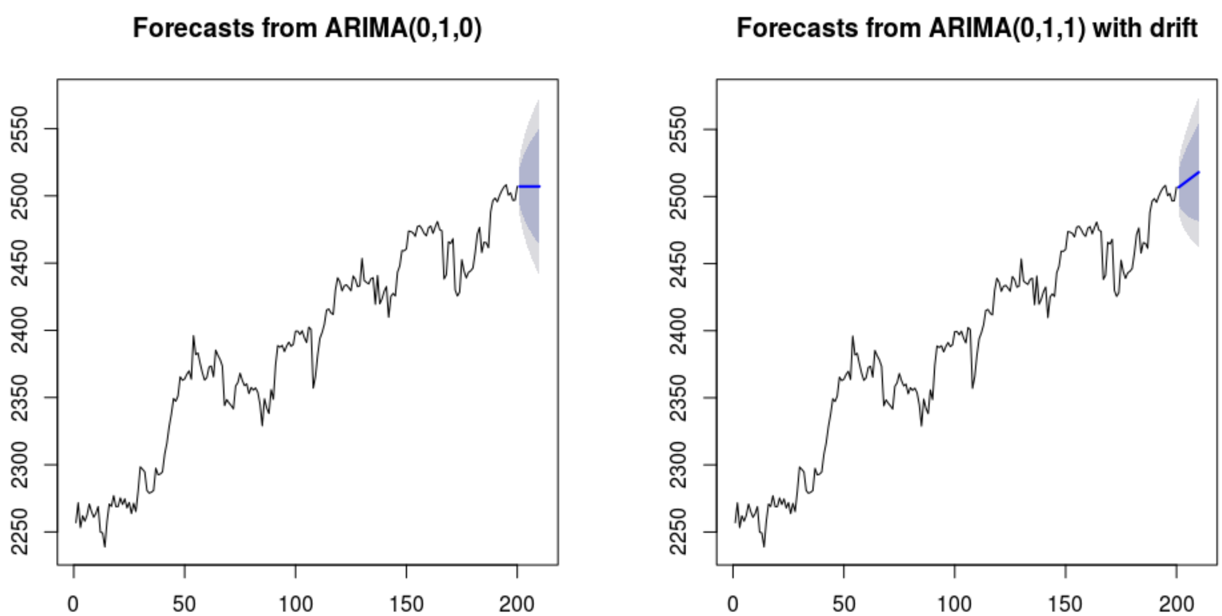



```
> tsdisplay(diff(prices.final)) # Taking first-order difference leads to stationarity.
```




```
> # To be frank, neither ACF nor PACF plots show any spikes whatsoever.
> my.arima.fit <- Arima(prices.final,c(0,1,0)) # So ARIMA(0,1,0) seems like a solid choice.
> summary(my.arima.fit)
ARIMA(0,1,0)
sigma^2 estimated as 114:  log likelihood=-753.64
AIC=1509.28   AICc=1509.3   BIC=1512.58
...
> auto.arima(prices.final,seasonal=F) # To compare one calls auto.arima on seas-adj data.
```

```
ARIMA(0,1,1) with drift                # It gives ARIMA(0,1,1) with a drift,
Coefficients:                          # DRIFT meaning that it has a constant NON-ZERO mean:
          ma1   drift                  # y_t = mu + theta_1 * eps_{t-1} + eps_t, where
      -0.1640  1.2410                  # mu != 0, eps_t ~ N(0,sigma^2)
s.e.   0.0777  0.6221
>  # So evidently auto.arima picks up on a constant drift mu = 1.25 of the differences,
>  # meaning that the original series (prior to differencing) is consistently growing.
>  # But unfortunately I won't cover ARIMAs with drift in much detail.
>
>  # Let's use both 1) our ARIMA(0,1,0), and 2) R's ARIMA(0,1,1) with drift, to forecast:
> par(mfrow=c(1,2))
> plot(forecast(my.arima.fit))
> plot(forecast(auto.arima(prices.final,seasonal=F)))
> par(mfrow=c(1,1))
```




```
> # One can see a considerable difference between forecasts
> # due to the DRIFT term (clear upward trend of predictions in the second plot).
```