Dev Template

Group 2

OLS Model: Paritioning the data by year

```r
ols.model1 <- function(data){
  lm(cnt ~ atemp + I(atemp^2) + holiday + weathersit, data)
}
```

The parameter estimates retain their signs if we split the dataset. From an inference POV, this is a good sign. Curiously enough, however, the actual parameter estimates change slightly.

Training on 2011 Data

```r
ols.model1(data2011)
```

```
##
## Call:
## lm(formula = cnt ~ atemp + I(atemp^2) + holiday + weathersit,
##     data = data)
##
## Coefficients:
## (Intercept)        atemp   I(atemp^2)  holidayTRUE  weathersit2  weathersit3
##   -1337.764      341.882       -4.811     -297.369     -548.976    -1915.115
```

Training on 2012 Data

```r
ols.model1(data2012)
```

```
##
## Call:
## lm(formula = cnt ~ atemp + I(atemp^2) + holiday + weathersit,
##     data = data)
##
## Coefficients:
## (Intercept)        atemp   I(atemp^2)  holidayTRUE  weathersit2  weathersit3
##   -2205.345      578.463       -9.004     -893.424     -818.524    -3127.477
```

Accounting for Growth in Bikeshare Users

There is a bit of a problem if we are to use the 2011 data as training data and 2012 as testing data. It assumes the number of users stays constant, which is certainly not true. **This is critical since our response variable is a count**. The number of bikesharing users has significantly grown since 2011 and continues to grow (could we cite something for this?) and treating the dataset as two disjoint datasets might be appopriate. Let's do a hypothesis test to see if $\mu_{2011} - \mu_{2012} = 0$.

```
t.test(data2011$cnt, data2012$cnt)
```

```
##
##  Welch Two Sample t-test
##
## data:  data2011$cnt and data2012$cnt
## t = -18.578, df = 685.5, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2426.069 -1962.276
## sample estimates:
## mean of x mean of y
##  3405.762  5599.934
```

A possible solution

As such, we need to account for this, since in theory, the true parameter estimate $\hat{\beta}_{atemp}$ should be independent of how many users there are and inturn which year it is. One possible solution is to consider scaling down our response variables in the 2012 data by the following factor $r = \frac{\mu_{2011}}{\mu_{2012}}$. In other words, we want to fit the scaled count variables $\tilde{c_i} = rc_i$. Only this way can we truly and fairly assess the parameter fit $\hat{\beta}$.

```
# Get the ratio of means scaling factor
r <- mean(data2011$cnt)/mean(data2012$cnt)
# Create adjusted count variable in 2012
data2012_ADJ <- data2012 %>% mutate(cnt = r * cnt)
```

Now, let us fit both models and see how the parameter estimates hold up. The parameter estimates indeed seem much more stable!

```
ols.model1(data2011)
```

```
##
## Call:
## lm(formula = cnt ~ atemp + I(atemp^2) + holiday + weathersit,
##     data = data)
##
## Coefficients:
## (Intercept)        atemp   I(atemp^2)  holidayTRUE  weathersit2  weathersit3
##   -1337.764      341.882       -4.811     -297.369     -548.976    -1915.115
```

```
ols.model1(data2012_ADJ)
```

```
##
## Call:
## lm(formula = cnt ~ atemp + I(atemp^2) + holiday + weathersit,
##     data = data)
##
## Coefficients:
## (Intercept)        atemp   I(atemp^2)  holidayTRUE  weathersit2  weathersit3
##   -1341.244      351.809       -5.476     -543.361     -497.809    -1902.065
```

Another way we could verify this is by using the adjusted count as a variable, reunifying the dataset, and fitting the same model.

```
ols.adjusted <- lm(cnt_adj ~ atemp + I(atemp^2) + holiday + weathersit, data = data)
ols.adjusted
```

```
##
## Call:
## lm(formula = cnt_adj ~ atemp + I(atemp^2) + holiday + weathersit,
##     data = data)
##
## Coefficients:
## (Intercept)        atemp   I(atemp^2)  holidayTRUE  weathersit2  weathersit3
##    -1331.42       345.36        -5.11      -433.37      -516.14     -1879.03
```

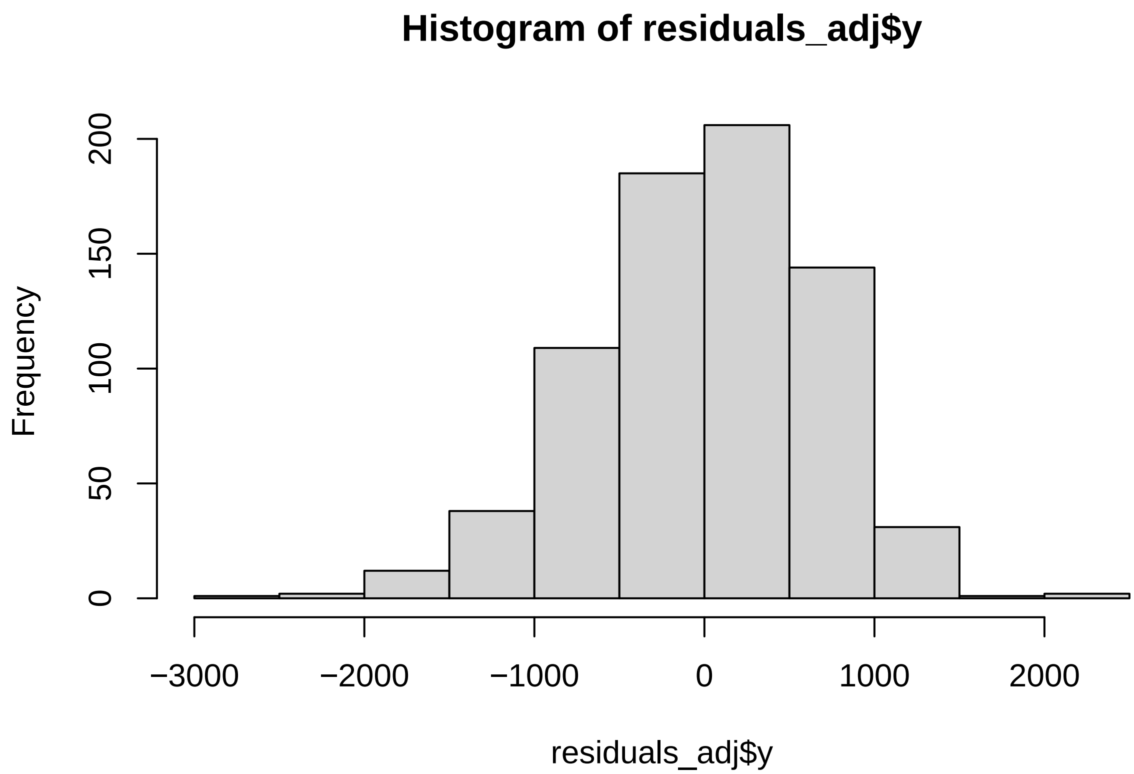
```
residuals_adj <- data.frame(x = data$instant, y = ols.adjusted$residuals)
# Residual plot by instance
p1 <- ggplot(data = residuals_adj) + geom_point(aes(x, y))
```

```
ols.unadjusted <- lm(cnt ~ yr + atemp + I(atemp^2) + holiday + weathersit, data = data)
ols.unadjusted
```

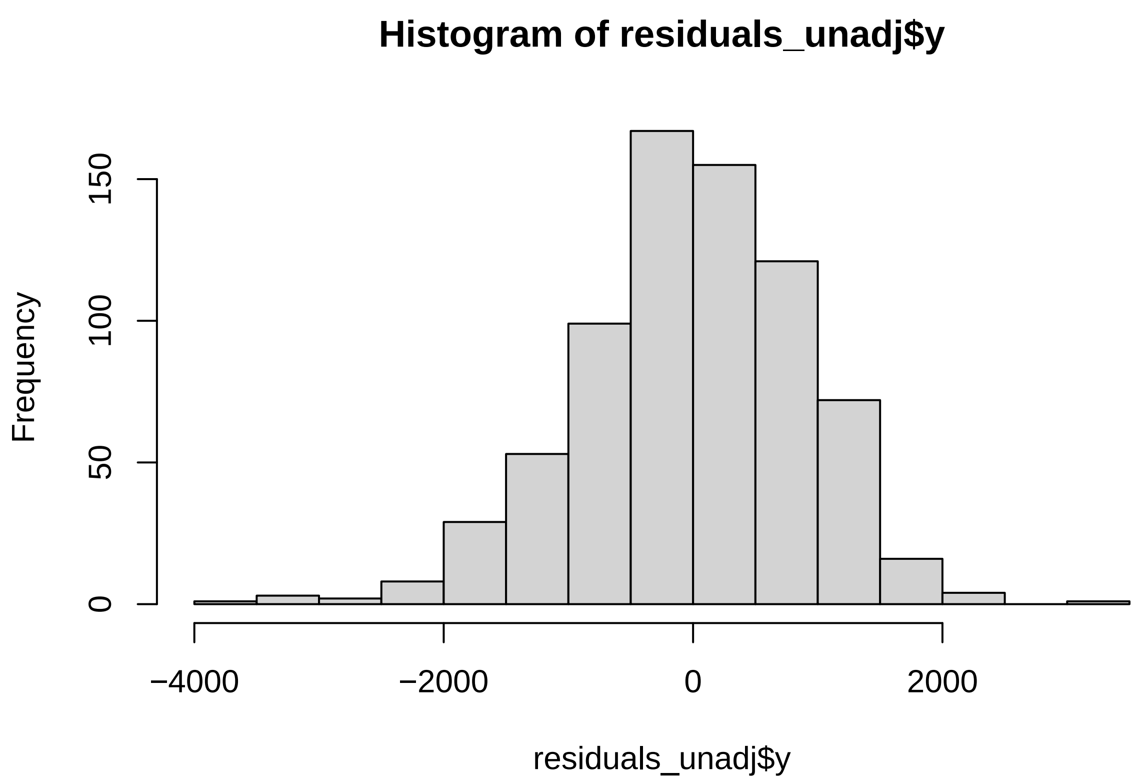
```
##
## Call:
## lm(formula = cnt ~ yr + atemp + I(atemp^2) + holiday + weathersit,
##     data = data)
##
## Coefficients:
## (Intercept)          yr1        atemp   I(atemp^2)  holidayTRUE  weathersit2
##   -2485.696     1973.774      439.458       -6.521     -640.302     -695.818
## weathersit3
##   -2343.449
```

```
residuals_unadj <- data.frame(x = data$instant, y = ols.unadjusted$residuals)
# Residual plot by instance
p2 <- ggplot(data = residuals_unadj) + geom_point(aes(x, y))
```

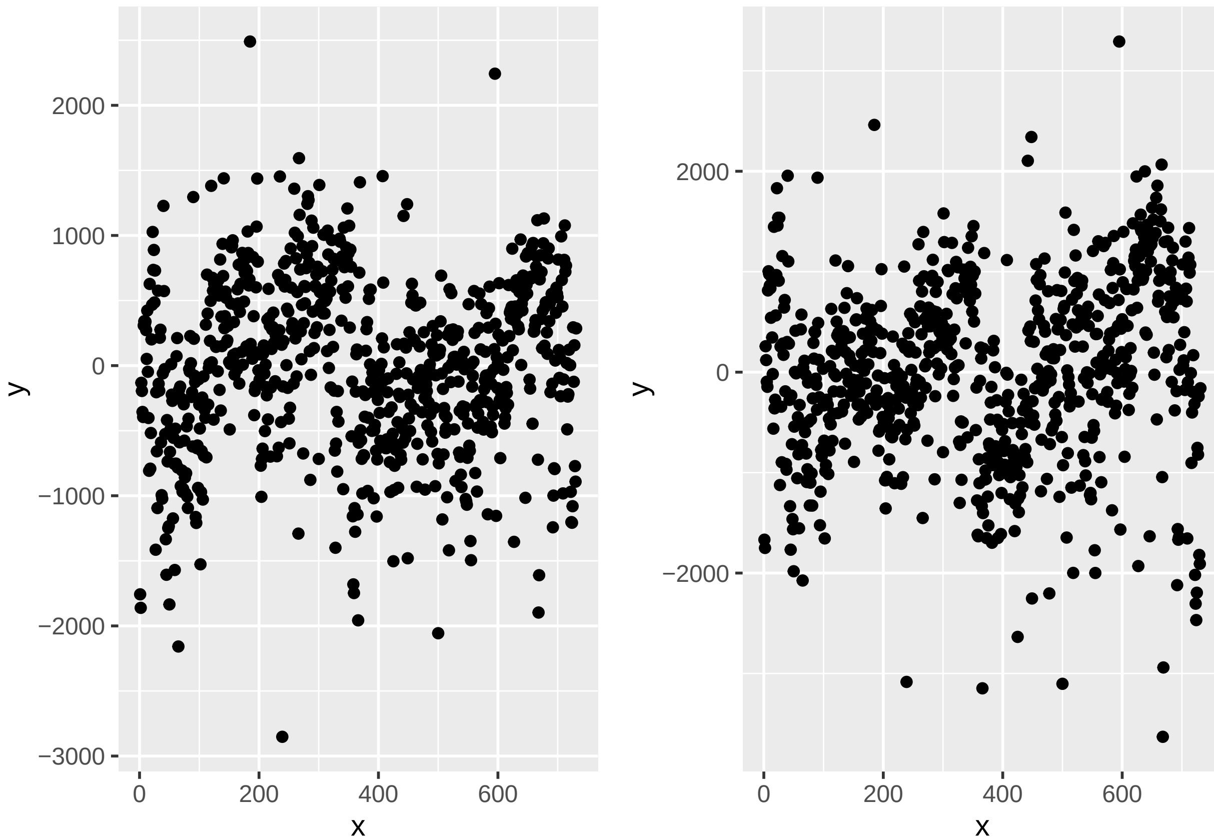
```
hist(residuals_adj$y)
```




```
hist(residuals_unadj$y)
```




```
plot_grid(p1, p2)
```


2000
1000
0
−1000
−2000
−3000
y
0
200
400
600
x
2000
0
−2000
y
0
200
400
600
x

Model Performance

```r
# Train model on 2011 data (don't show it 2012 data)
ols.model <- ols.model1(data = data2011)
#summary(ols.model)
```

```r
preds2012 <- predict(ols.model, data2012)
```

MSE of unadjusted response on 2012 dataset (training on 2011 data only)

```r
# MSEs
MSE_unadj <- sqrt(mean((preds2012 - data2012$cnt)^2))
MSE_unadj
```

```
## [1] 2290.639
```

MSE of adjusted response on 2012 dataset (training on 2011 data only)

```r
MSE_adj <- sqrt(mean((preds2012 - data2012$cnt_adj)^2))
MSE_adj
```

```
## [1] 653.3103
```

In conclusion, adjusting the response variable shows that the model performs better than initially thought. By adjusting for an increasing base of bike share users, the parameter β is "given a fair shot" to estimate the effects our predictors have since the adjusted response variable "relevels" the playing field.

Now lets compare it to if we used the `yr` variable as a predictor, how does it compare? In theory, we have the same exact information, so we expect the model performance on the testing data to not be too disparate. Why would changing from a indicator variable scheme to a scaled response variable scheme change performance if both use the same "information"?

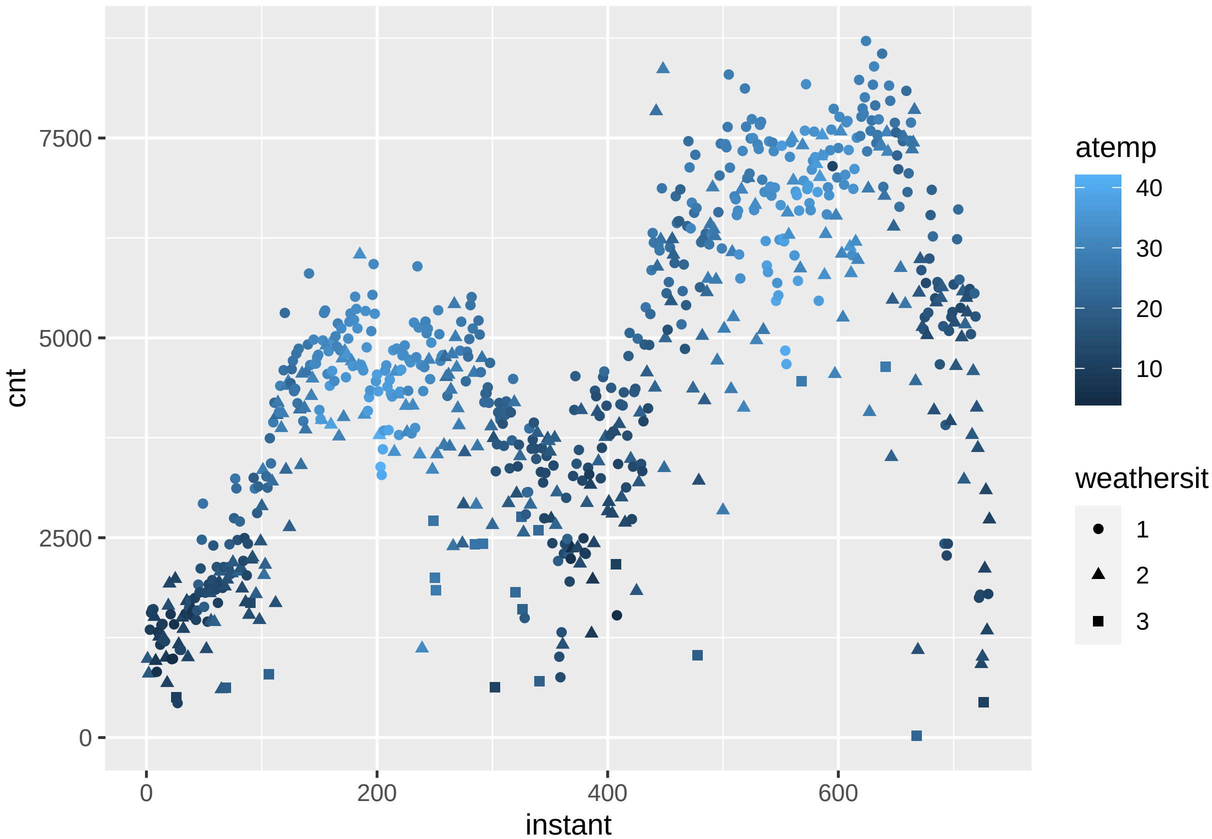
Surprisingly, the model with the scaled response variable scheme did perform better. So, if we account the response variable for expected growth r rather than add a constant $\Delta\mu$ to our fitted values (the effect of regressing an indicator variable), this indicates we might get better model performance.

```r
preds2012 <- predict(ols.unadjusted, data2012)
MSE <- sqrt(mean((preds2012 - data2012$cnt)^2))
MSE
```

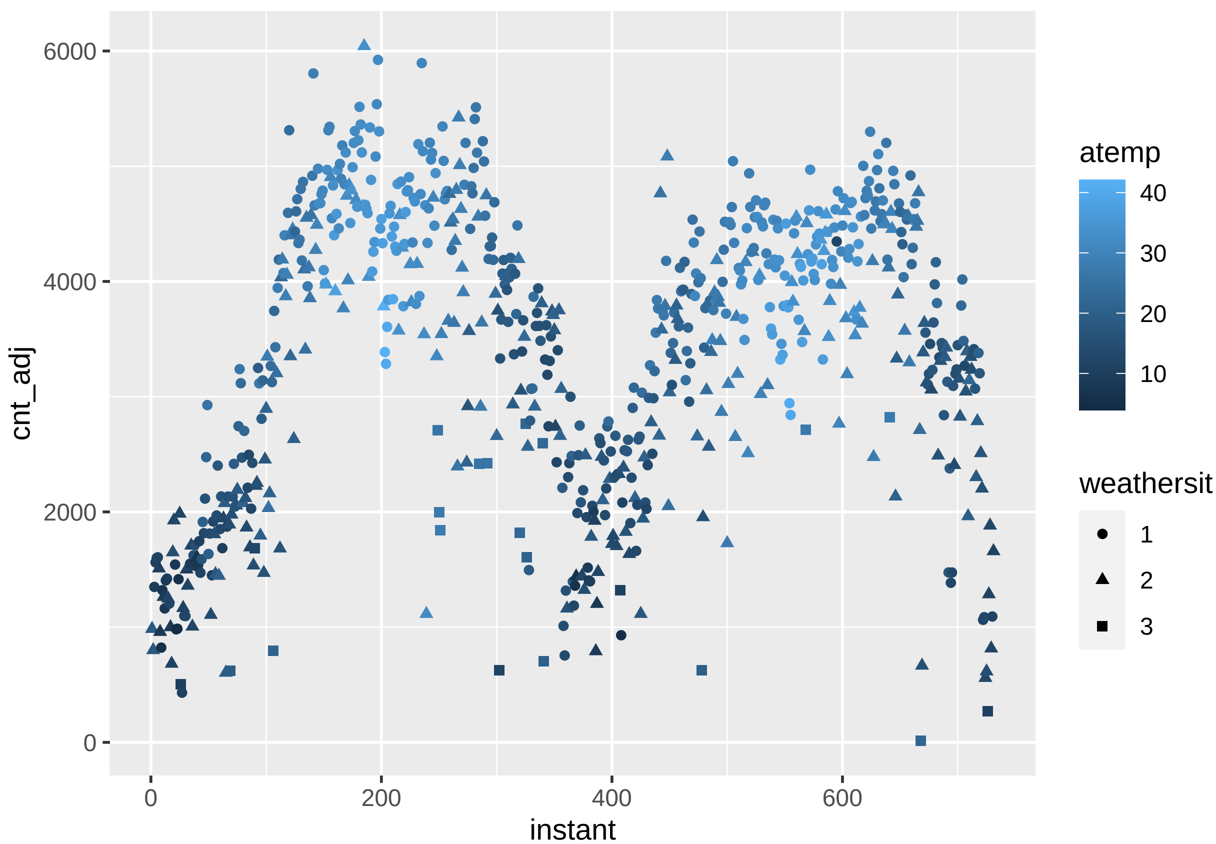
```
## [1] 1022.831
```

Plot showing the scaled and unscaled data series

```
rbind(data2011, data2012) %>%
  ggplot() +
  geom_point(aes(x = instant, y = cnt, color = atemp, shape = weathersit))
```




```
data %>%
  ggplot() +
  geom_point(aes(x = instant, y = cnt_adj, color = atemp, shape = weathersit))
```



6000
4000
2000
0
cnt_adj
0
200
400
600
instant
atemp
40
30
20
10
weathersit
1
2
3