LAX Ground Vehicle Traffic Volume Forecasting

Brady Chiu

August 07, 2016

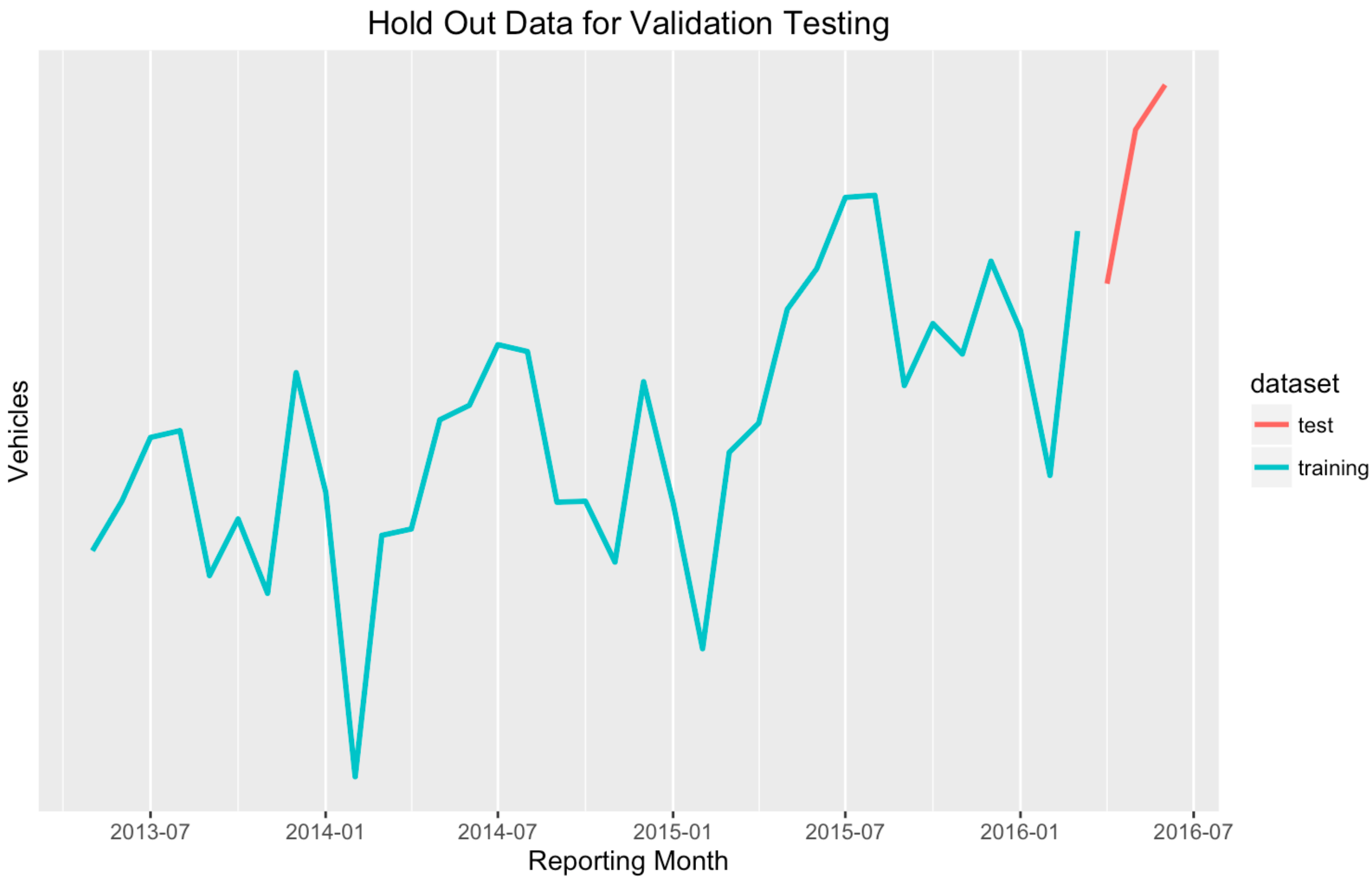
1 TL;DR

- **Objective:** Forecast LAX ground vehicle traffic volume for the year of 2017
- **Apporach:** We assessed 3 forecasting models: STL Decomposition, Holt-Winters Exponential Smoothing, and Arima
- **Finding:** STL Decomposition forecasting was most accurate, predicting ~40MM vehicles will pass through LAX in 2017

2 Setup

2.1 Validation Data

To setup the data for our analysis, we'll take the last 3 months and set them aside for validation. We'll use the remaining data to train our models.

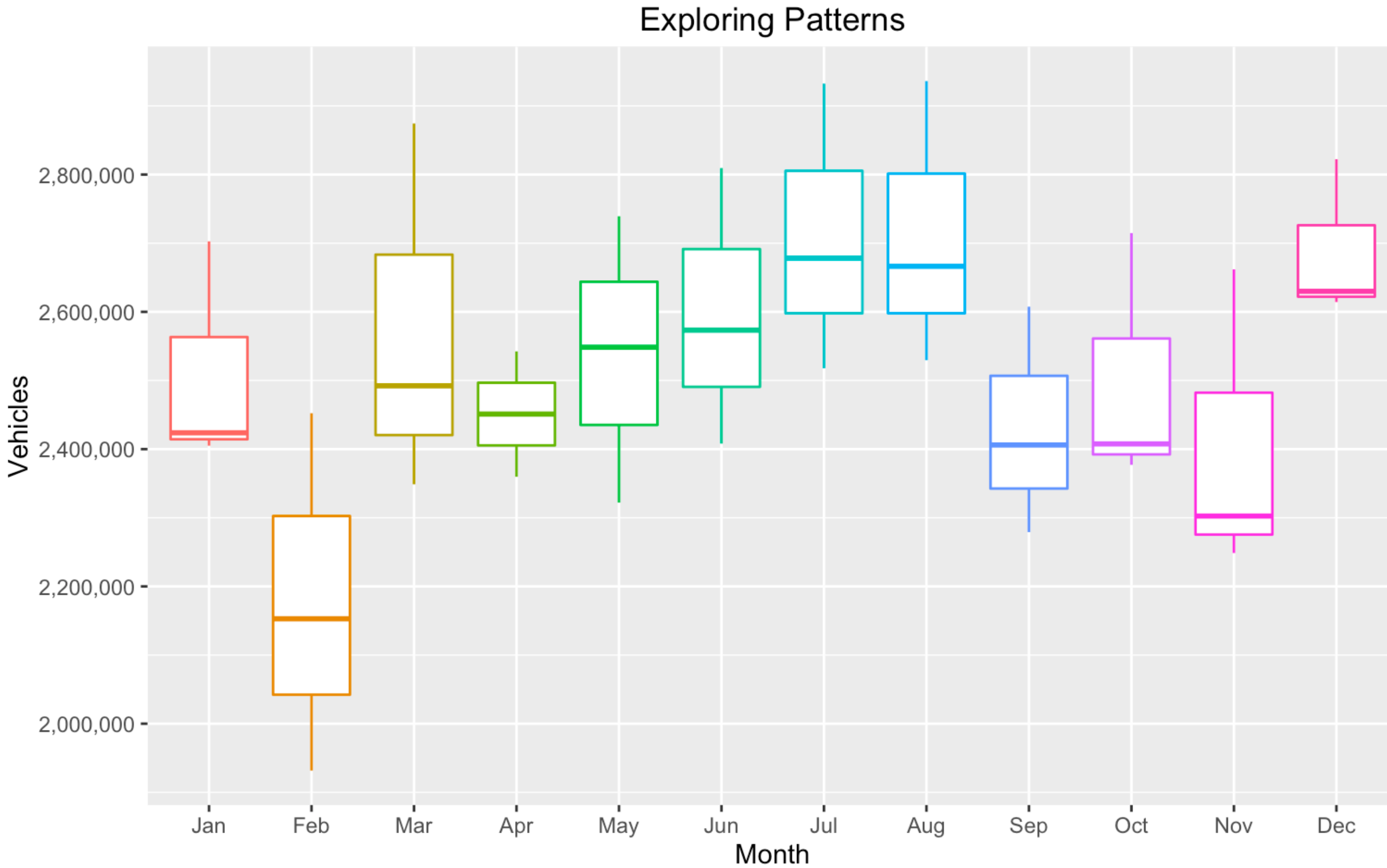



```
##          Jan     Feb     Mar     Apr     May     Jun     Jul     Aug
## 2013                                 2322085 2408061 2517892 2529552
## 2014 2423773 1931681 2348707 2359741 2548453 2573367 2678236 2666378
## 2015 2404930 2152837 2492211 2542424 2739284 2809522 2932503 2936218
## 2016 2702732 2452320 2874440
##          Sep     Oct     Nov     Dec
## 2013 2279129 2377058 2248586 2629896
## 2014 2406043 2407597 2302482 2614235
## 2015 2607537 2714757 2661911 2822376
## 2016
```

2.2 Exploring Patterns

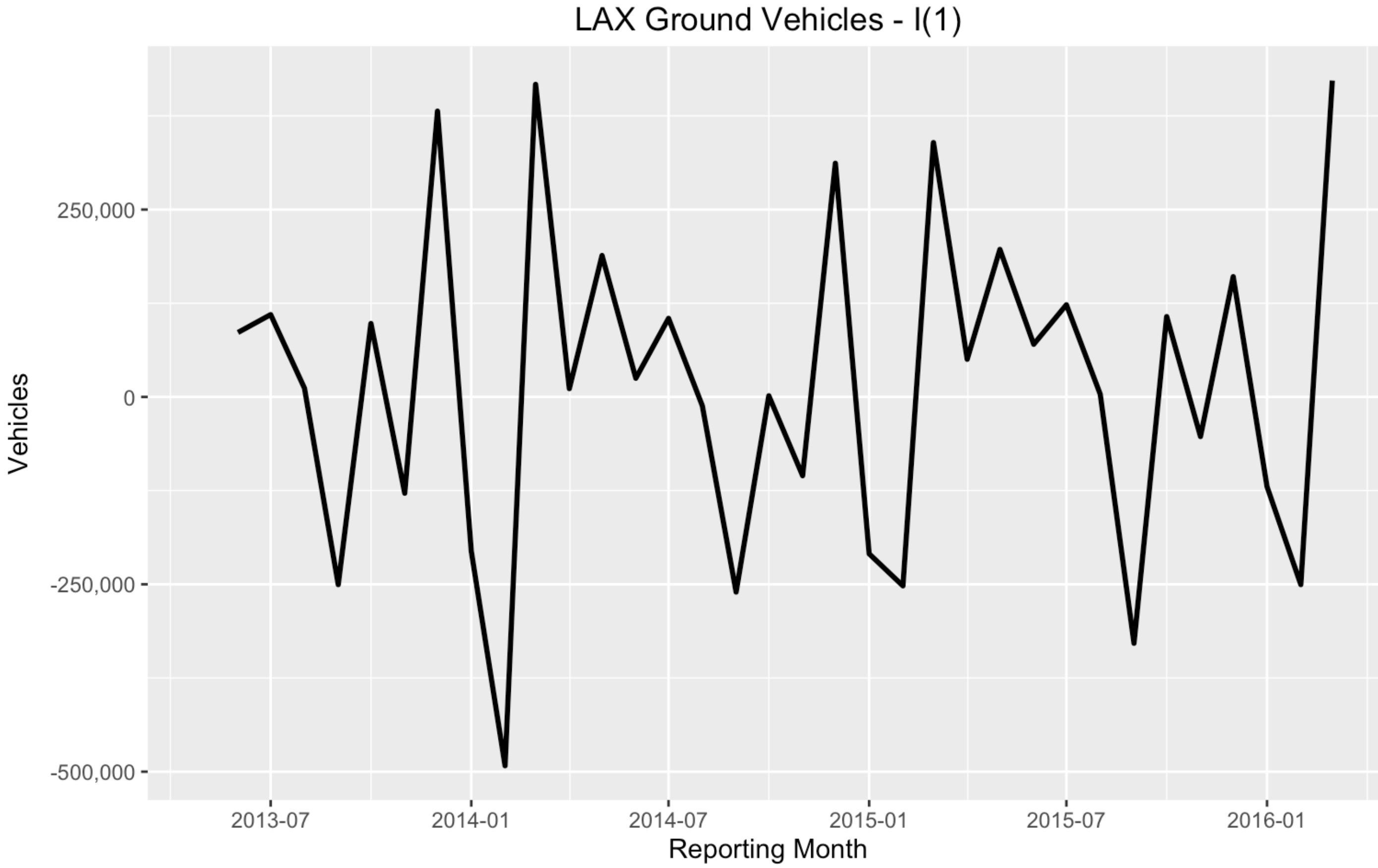
As expected with travel patterns, we see relatively strong 12 month seasonality:

- February is consistently the lowest month
- July / August / December are consistently the highest months
- Dips consistently occur during September / October / November periods



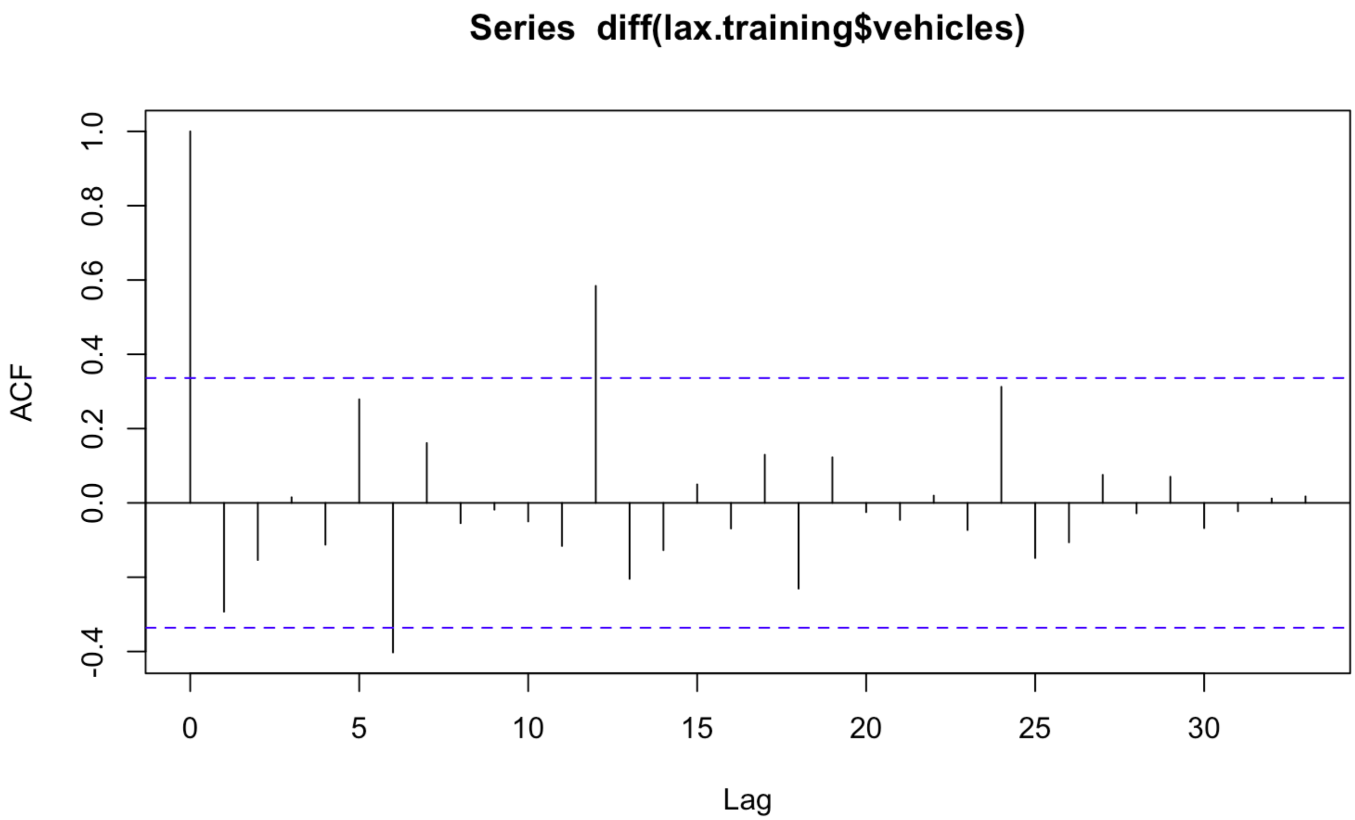
2.3 Stationary

Before proceeding, we'll need to make our data stationary over time, so we'll start with an integrated of order 1 and see if it becomes stationary.



2.3.1 ACF

Our Autocorrelation Function (ACF) shows significance at lag 12 and almost significance at lag 24. This makes intuitive sense as travel data is seasonal on an annual level.



2.3.2 Ljung-Box

Our Ljung-Box test shows statistical significance, validating I(1) is stationary.

```
## 
##  Box-Ljung test
## 
## data:  diff(lax.training$vehicles)
## X-squared = 46.621, df = 20, p-value = 0.0006619
```

2.3.3 ADF

Our Augmented Dickey-Fuller (ADF) test shows statistical significance at a p-value < 0.05, validating I(1) is stationary.

```
## 
##  Augmented Dickey-Fuller Test
## 
## data:  diff(lax.training$vehicles)
## Dickey-Fuller = -4.0279, Lag order = 3, p-value = 0.02033
## alternative hypothesis: stationary
```

2.3.4 KPSS

Our Kwiatkowski-Phillips-Schmidt-Shin (KPSS) test shows no statistical significance at a p-value < 0.05, validating I(1) is stationary.

```
##
##  KPSS Test for Level Stationarity
##
## data:  diff(lax.training$vehicles)
## KPSS Level = 0.034776, Truncation lag parameter = 1, p-value = 0.1
```

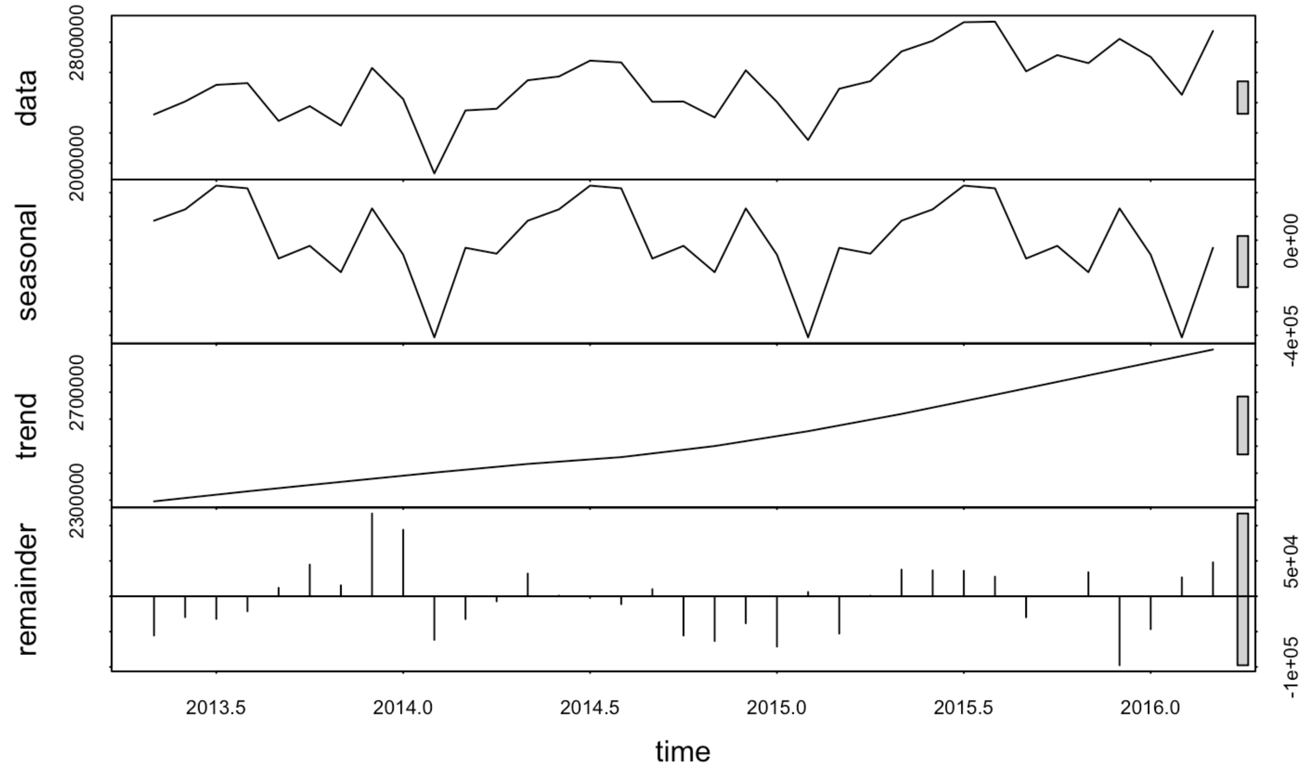
3 Model Selection

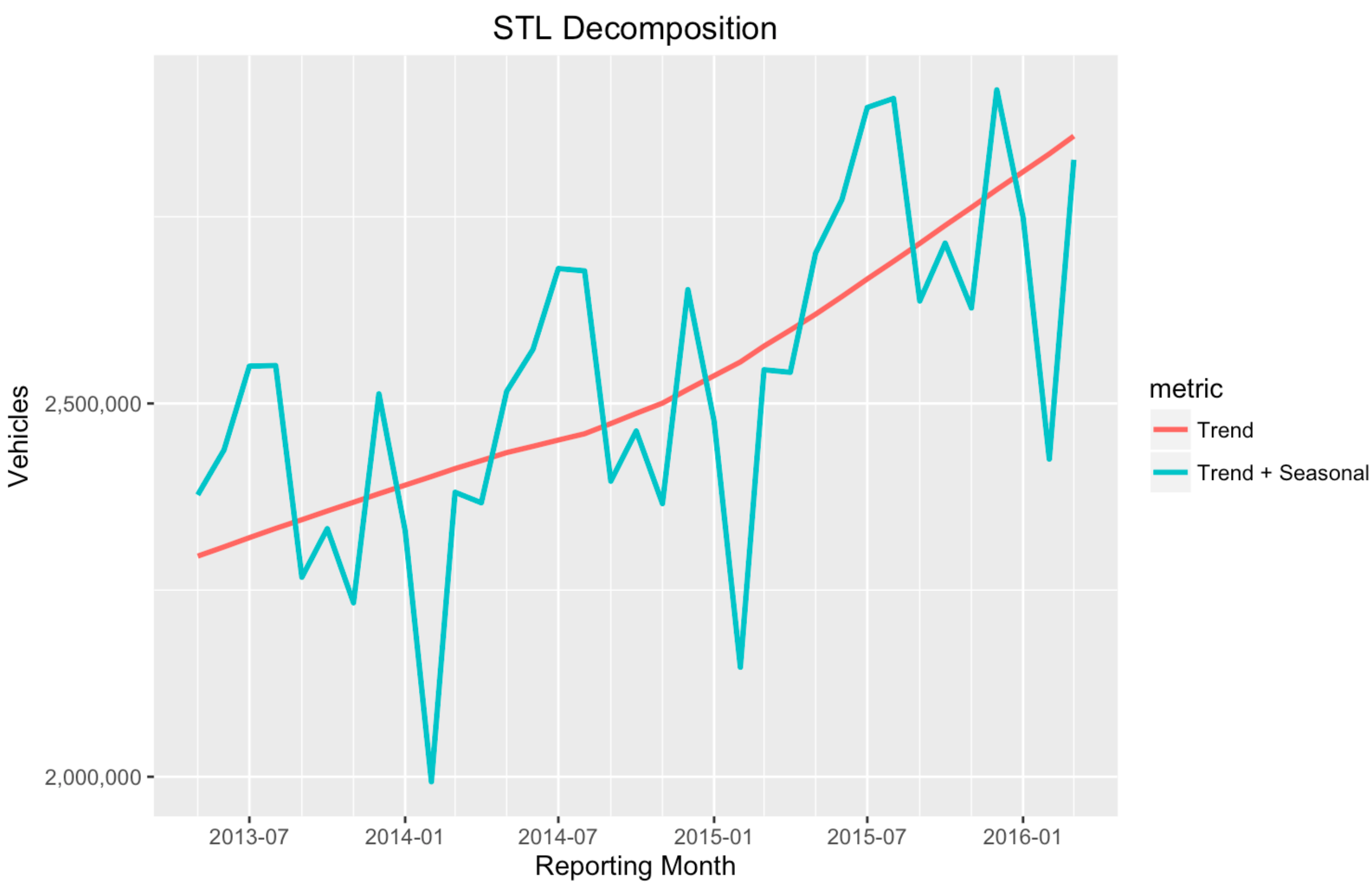
We'll run forecasting using our three models:

- STL Decomposition
- Holt-Winters Exponential Smoothing
- Arima

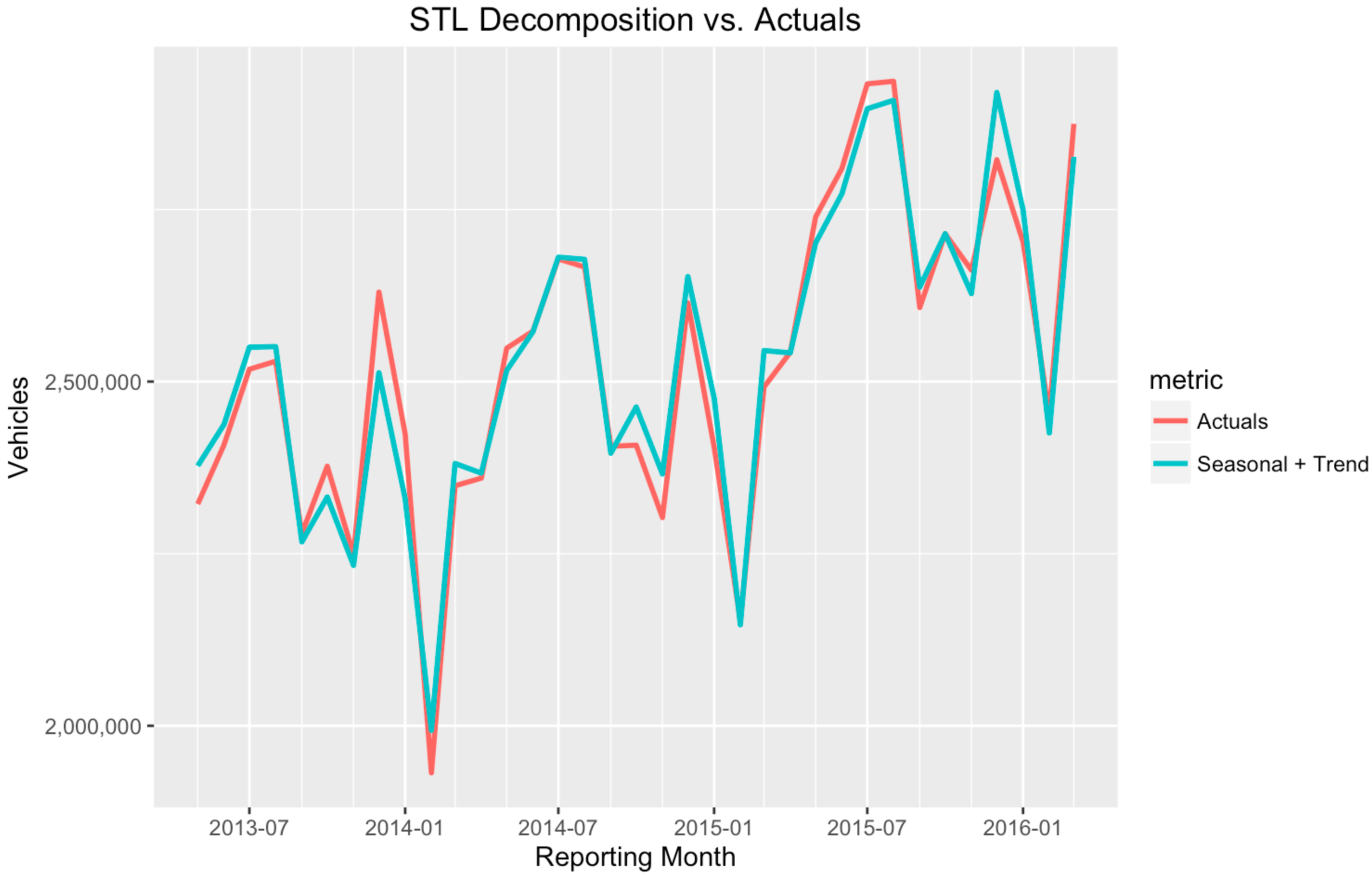
3.1 STL Decomposition

3.1.1 Analysis

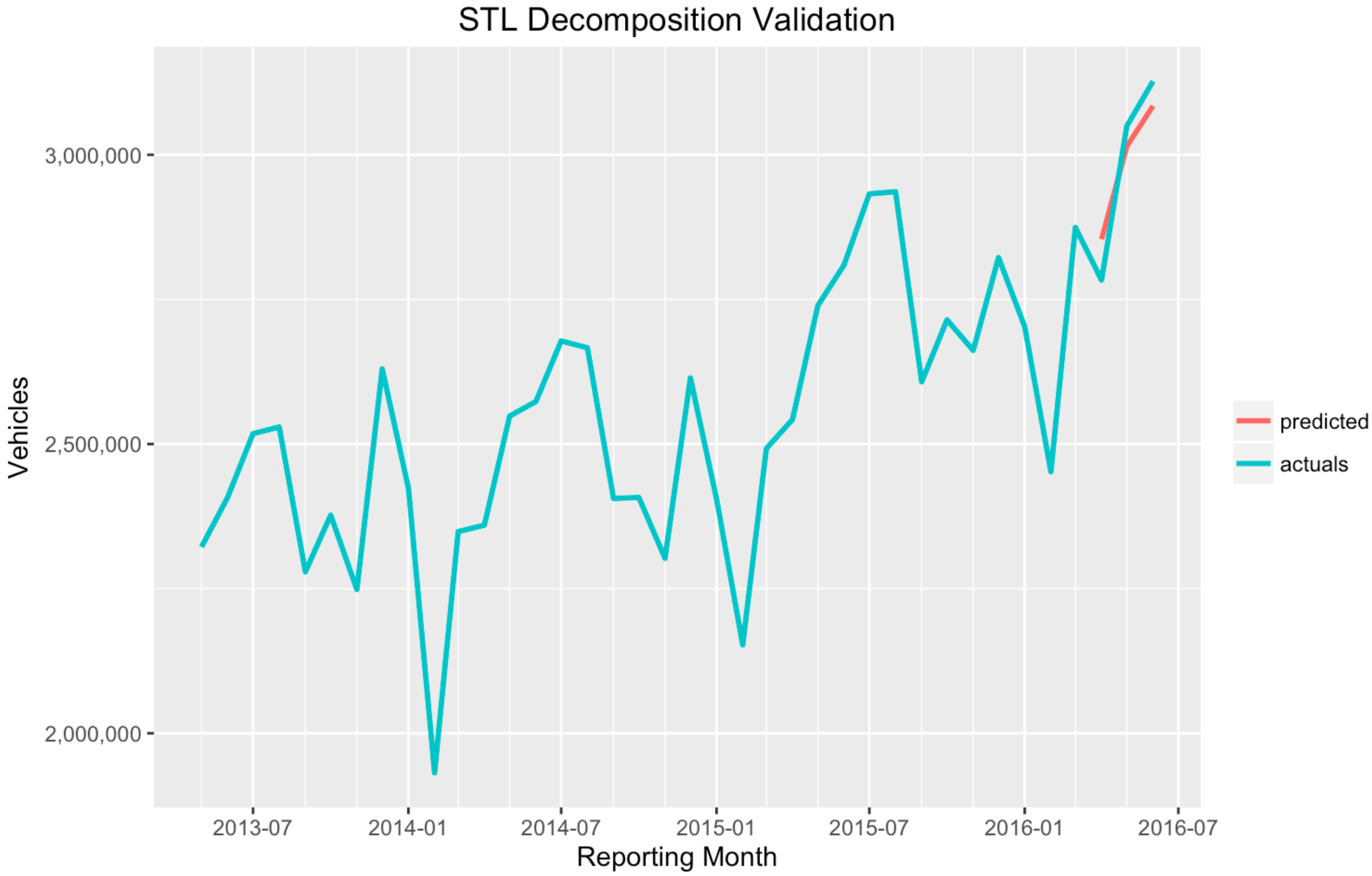




3.1.2 Validation



	Point.Forecast	Lo.80	Hi.80	Lo.95	Hi.95
Apr 2016	2,854,059	2,781,038	2,927,080	2,742,384	2,965,734
May 2016	3,014,717	2,927,119	3,102,316	2,880,747	3,148,687
Jun 2016	3,084,528	2,984,273	3,184,783	2,931,202	3,237,855



model	mae	mape	mse	mpe
STL	49,374.81	0.0168	2,672,397,950	-1e-04

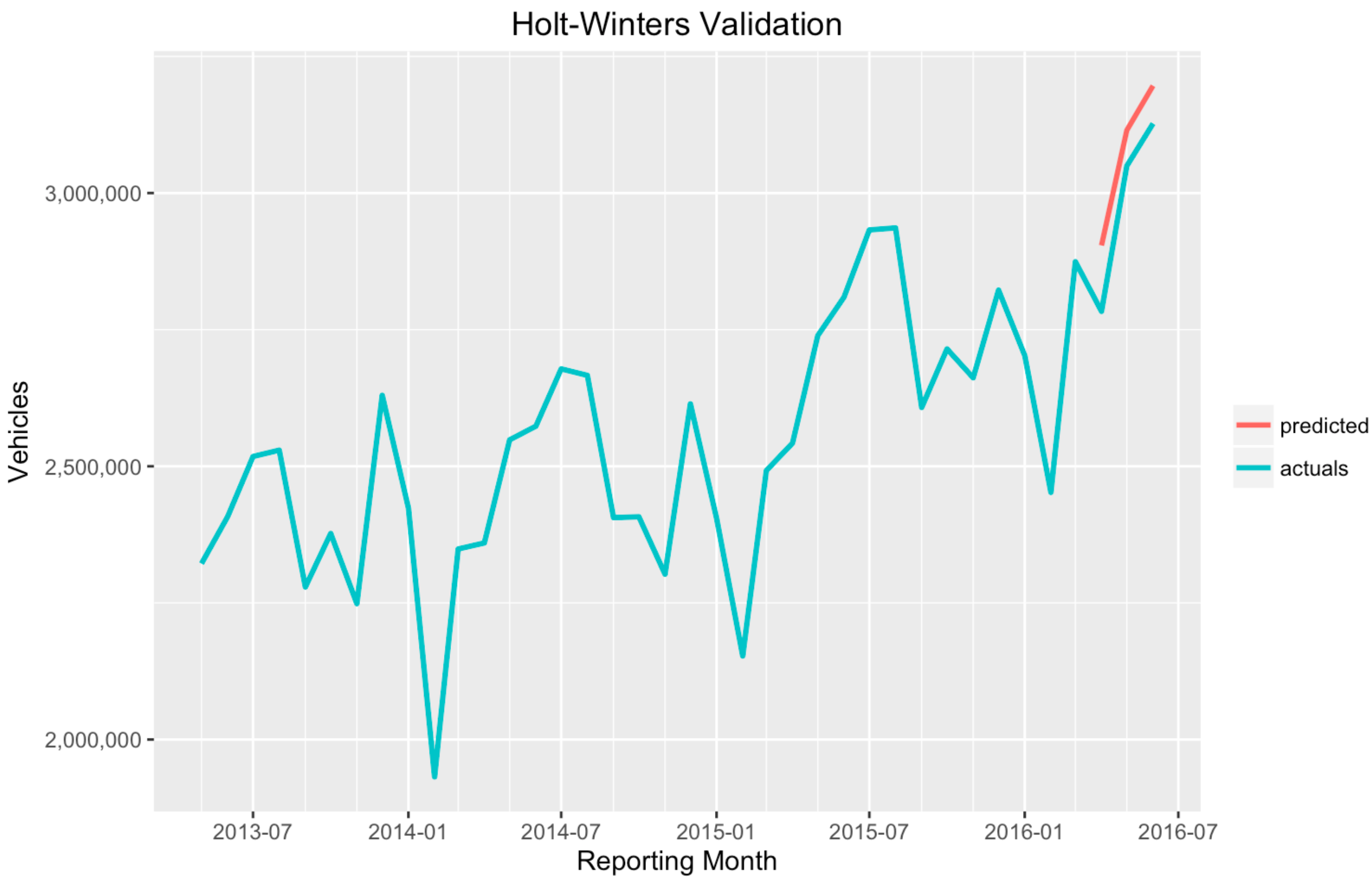
3.2 Holt-Winters Exponential Smoothing

3.2.1 Analysis

```
## Holt-Winters exponential smoothing with trend and additive seasonal component.
##
## Call:
## HoltWinters(x = lax.ts12)
##
## Smoothing parameters:
##  alpha: 0.0466399
##  beta : 1
##  gamma: 1
##
## Coefficients:
##            [,1]
## a   2864807.764
## b     33785.349
## s1     5489.864
## s2   182580.965
## s3   230187.267
## s4   327370.776
## s5   302049.128
## s6   -54961.301
## s7    20208.747
## s8   -69865.707
## s9    56780.354
## s10  -96252.978
## s11 -378702.415
## s12    9632.236
```

3.2.2 Validation

	Point.Forecast	Lo.80	Hi.80	Lo.95	Hi.95
Apr 2016	2,904,083	2,820,378	2,987,788	2,776,068	3,032,098
May 2016	3,114,959	3,030,891	3,199,028	2,986,388	3,243,530
Jun 2016	3,196,351	3,111,471	3,281,231	3,066,538	3,326,164



model	mae	mape	mse	mpe
Holt-Winters	85,048.83	0.029	7,868,504,847	-0.029

3.3 Arima

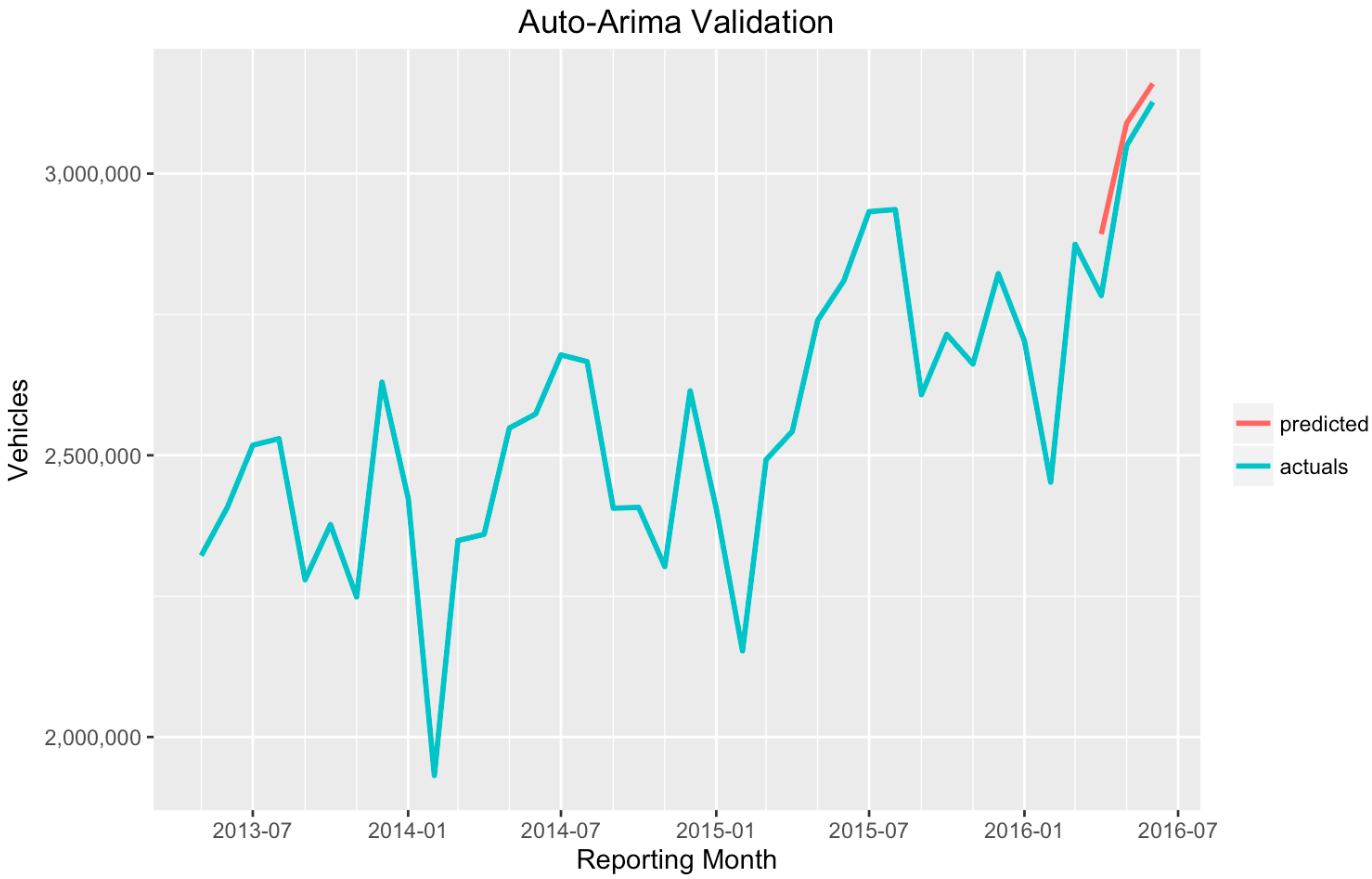
3.3.1 Analysis

```
## Series: lax.ts12
## ARIMA(0,1,1)(0,1,0)[12]
##
## Coefficients:
##           ma1
##       -0.3585
## s.e.   0.1903
##
## sigma^2 estimated as 5.971e+09:  log likelihood=-278.38
## AIC=560.76   AICc=561.39   BIC=562.94
```

3.3.2 Validation

	Point.Forecast	Lo.80	Hi.80	Lo.95	Hi.95
Apr 2016	2,892,655	2,793,625	2,991,684	2,741,202	3,044,108
May 2016	3,089,515	2,971,858	3,207,172	2,909,574	3,269,456

Jun 2016	3,159,753	3,026,039	3,293,467	2,955,255	3,364,251



model	mae	mape	mse	mpe
Arima	60,558.61	0.0209	4,858,125,254	-0.0209

3.4 Model Comparison

We compared our models using the following error metrics:

- Mean Absolute Error
- Mean Absolute Percentage Error
- Mean Squared Error
- Mean Percentage Error

model	mae map	e	mse m	pe
STL 49,	374.81 0.0	2 2,6	72,397,950 0.	00
Holt-Winters	85,048.83	0.03	7,868,504,847	-0.03
Arima 60,	558.61 0.0	2 4,8	58,125,254 -0.	02

Our STL Decomposition model performed the best

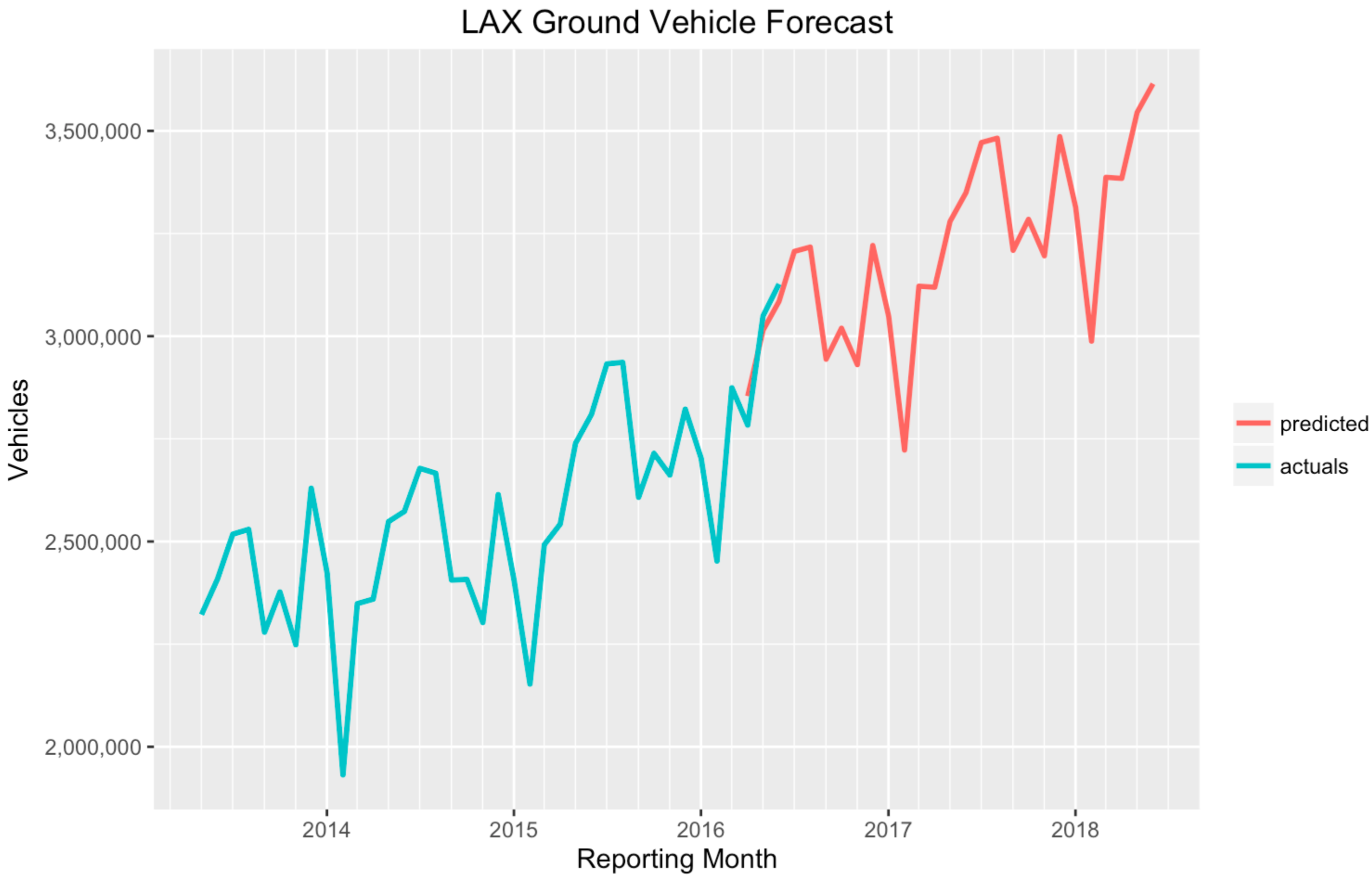
4 Forecasting

We'll use our STL Decomposition to project LAX ground vehicles 2 years out

	Point.Forecast	Lo.80	Hi.80	Lo.95	Hi.95
Apr 2016	2,854,059	2,781,038	2,927,080	2,742,384	2,965,734
May 2016	3,014,717	2,927,119	3,102,316	2,880,747	3,148,687
Jun 2016	3,084,528	2,984,273	3,184,783	2,931,202	3,237,855
Jul 2016	3,206,524	3,094,877	3,318,171	3,035,774	3,377,273
Aug 2016	3,217,131	3,095,001	3,339,261	3,030,350	3,403,913
Sep 2016	2,943,955	2,812,033	3,075,876	2,742,198	3,145,711
Oct 2016	3,019,492	2,878,325	3,160,659	2,803,595	3,235,389
Nov 2016	2,930,651	2,780,682	3,080,619	2,701,294	3,160,008
Dec 2016	3,220,991	3,062,591	3,379,392	2,978,738	3,463,244
Jan 2017	3,048,465	2,881,945	3,214,985	2,793,794	3,303,135
Feb 2017	2,722,763	2,548,392	2,897,135	2,456,085	2,989,442
Mar 2017	3,121,837	2,939,846	3,303,828	2,843,505	3,400,168
Apr 2017	3,119,221	2,929,815	3,308,628	2,829,549	3,408,894
May 2017	3,279,880	3,083,238	3,476,522	2,979,142	3,580,618
Jun 2017	3,349,691	3,145,973	3,553,408	3,038,132	3,661,250
Jul 2017	3,471,686	3,261,036	3,682,336	3,149,525	3,793,847
Aug 2017	3,482,294	3,264,841	3,699,747	3,149,728	3,814,859
Sep 2017	3,209,117	2,984,977	3,433,257	2,866,325	3,551,909
Oct 2017	3,284,654	3,053,934	3,515,375	2,931,798	3,637,511
Nov 2017	3,195,813	2,958,608	3,433,018	2,833,039	3,558,587
Dec 2017	3,486,154	3,242,552	3,729,755	3,113,598	3,858,710
Jan 2018	3,313,627	3,063,710	3,563,544	2,931,413	3,695,842
Feb 2018	2,987,926	2,731,768	3,244,084	2,596,166	3,379,685
Mar 2018	3,386,999	3,124,669	3,649,329	2,985,799	3,788,199
Apr 2018	3,384,384	3,115,944	3,652,824	2,973,841	3,794,927
May 2018	3,545,042	3,270,551	3,819,533	3,125,245	3,964,840
Jun 2018	3,614,853	3,334,366	3,895,341	3,185,885	4,043,822

reporting_month	value
2016-04-01	2,854,059

	2016-05-01	3,014,717
	2016-06-01	3,084,528
	2016-07-01	3,206,524
	2016-08-01	3,217,131
	2016-09-01	2,943,955
	2016-10-01	3,019,492
	2016-11-01	2,930,651
	2016-12-01	3,220,991
	2017-01-01	3,048,465
	2017-02-01	2,722,763
	2017-03-01	3,121,837
	2017-04-01	3,119,221
	2017-05-01	3,279,880
	2017-06-01	3,349,691
	2017-07-01	3,471,686
	2017-08-01	3,482,294
	2017-09-01	3,209,117
	2017-10-01	3,284,654
	2017-11-01	3,195,813
	2017-12-01	3,486,154
	2018-01-01	3,313,627
	2018-02-01	2,987,926
	2018-03-01	3,386,999
	2018-04-01	3,384,384
	2018-05-01	3,545,042
	2018-06-01	3,614,853

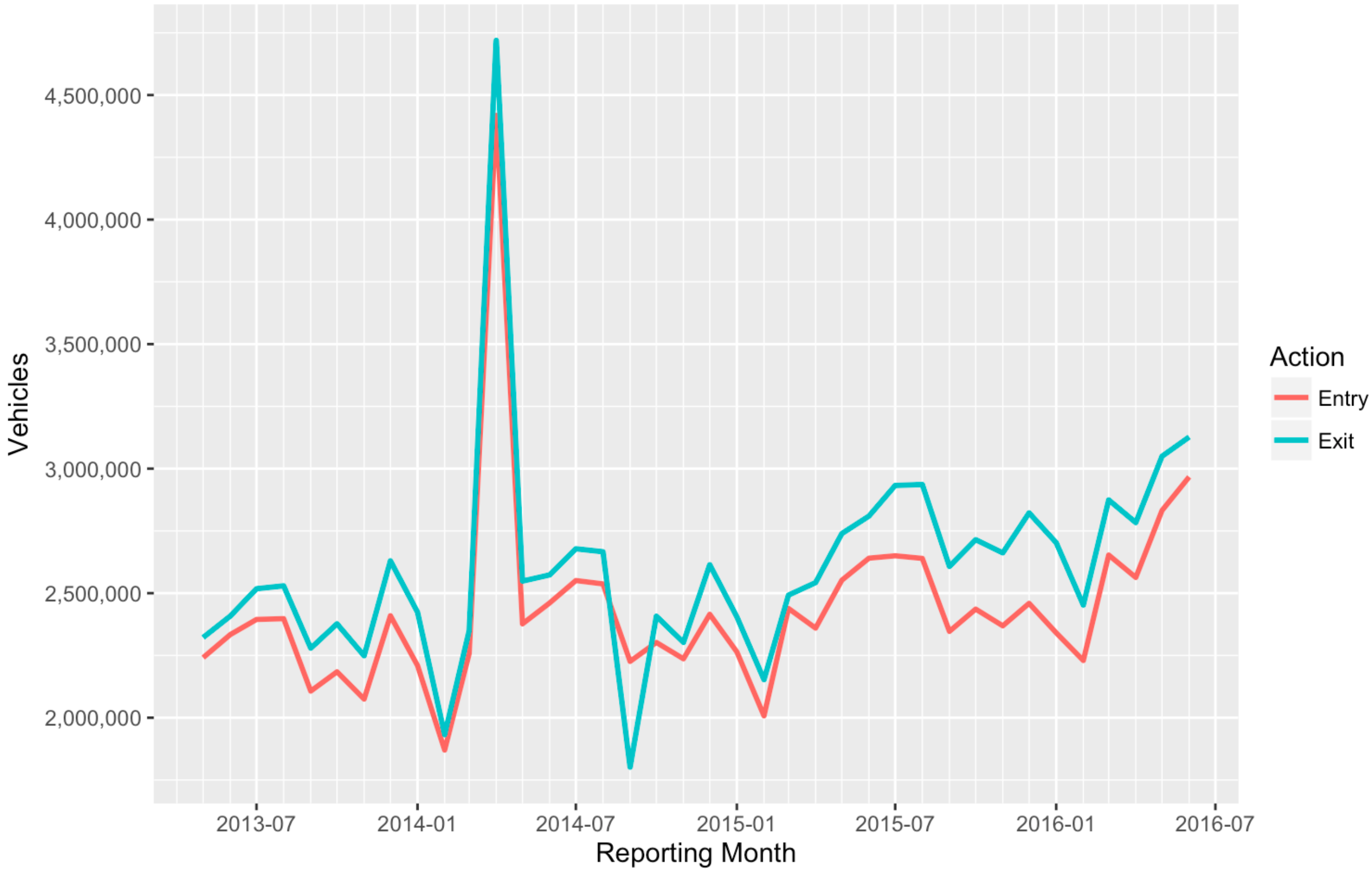


5 Appendix

5.1 Data Source

Our raw data for LAX Ground Vehicle Traffic Volume came from www.data.gov. See here (https://catalog.data.gov/dataset/los-angeles-international-airport-ground-vehicle-traffic-volume-cb231) for the original data source.

5.2 Data Quality Check



We see three odd things about our data, and will do what we can to fix them:

1. Month of 2014-04 has an unusual spike in trip volume
2. Month of 2014-09 is the only month where entries out pace exists
3. Exits outpace entries every month

5.2.1 Part 1

Month of 2014-04 has an unusual spike in trip volume.

reporting_month	action	level	vehicles
2014-04-01	Entry	Lower Level	1,027,606
2014-04-01	Entry	Lower Level	1,027,606
2014-04-01	Entry	Upper Level	1,181,556
2014-04-01	Entry	Upper Level	1,181,556
2014-04-01	Exit	Lower Level	1,691,869
2014-04-01	Exit	Lower Level	1,691,869
2014-04-01	Exit	Upper Level	667,872
2014-04-01	Exit	Upper Level	667,872

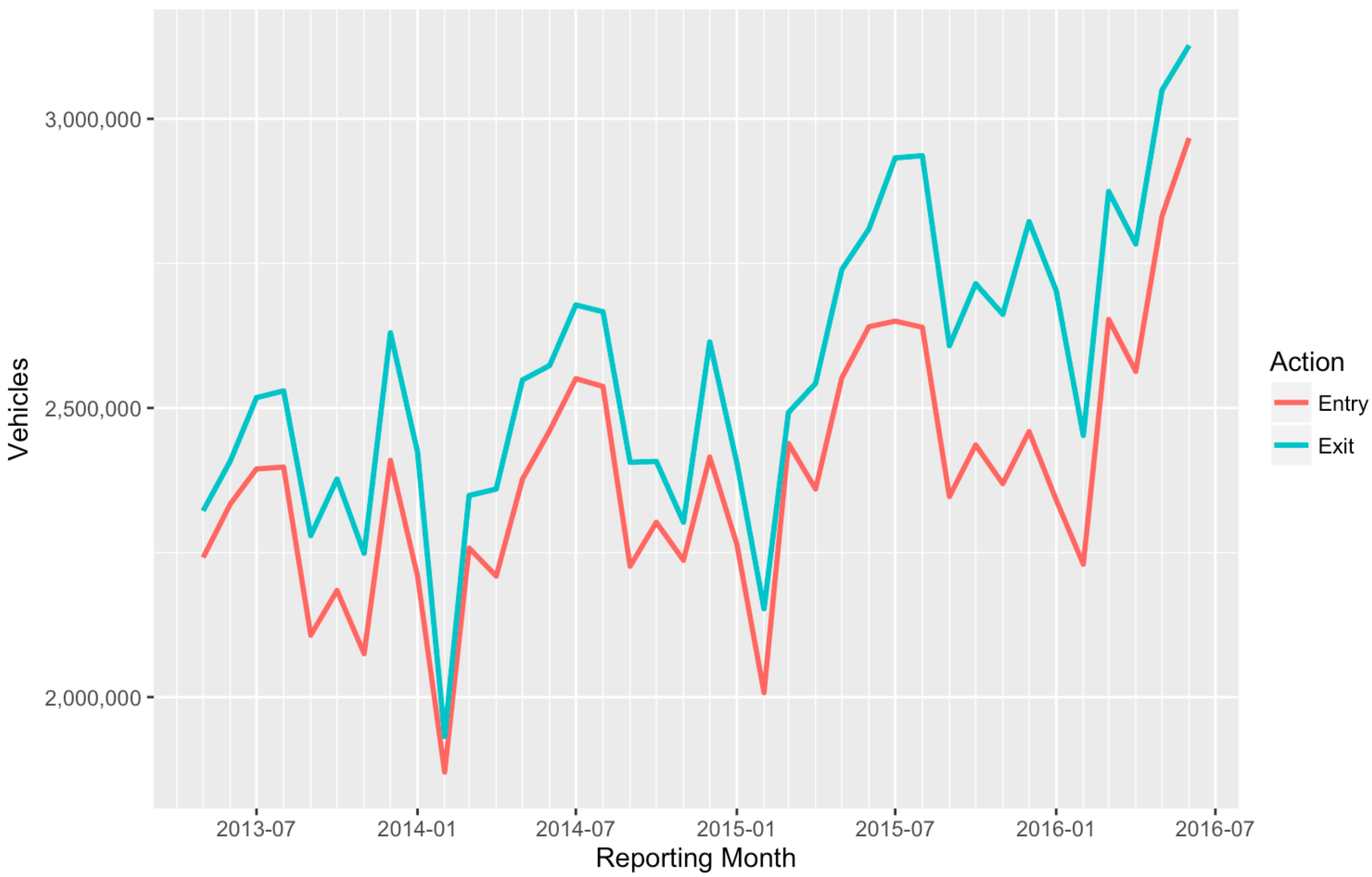
Investigating our raw data shows dup rows for this month, so we’ll go ahead and dedup rows for that month.

reporting_month	action	level	vehicles
2014-04-01	Entry	Lower Level	1,027,606
2014-04-01	Entry	Upper Level	1,181,556
2014-04-01	Exit	Lower Level	1,691,869
2014-04-01	Exit	Upper Level	667,872

5.2.2 Part 2

We see month of 2014-09 is the only month where entries out pace exists.

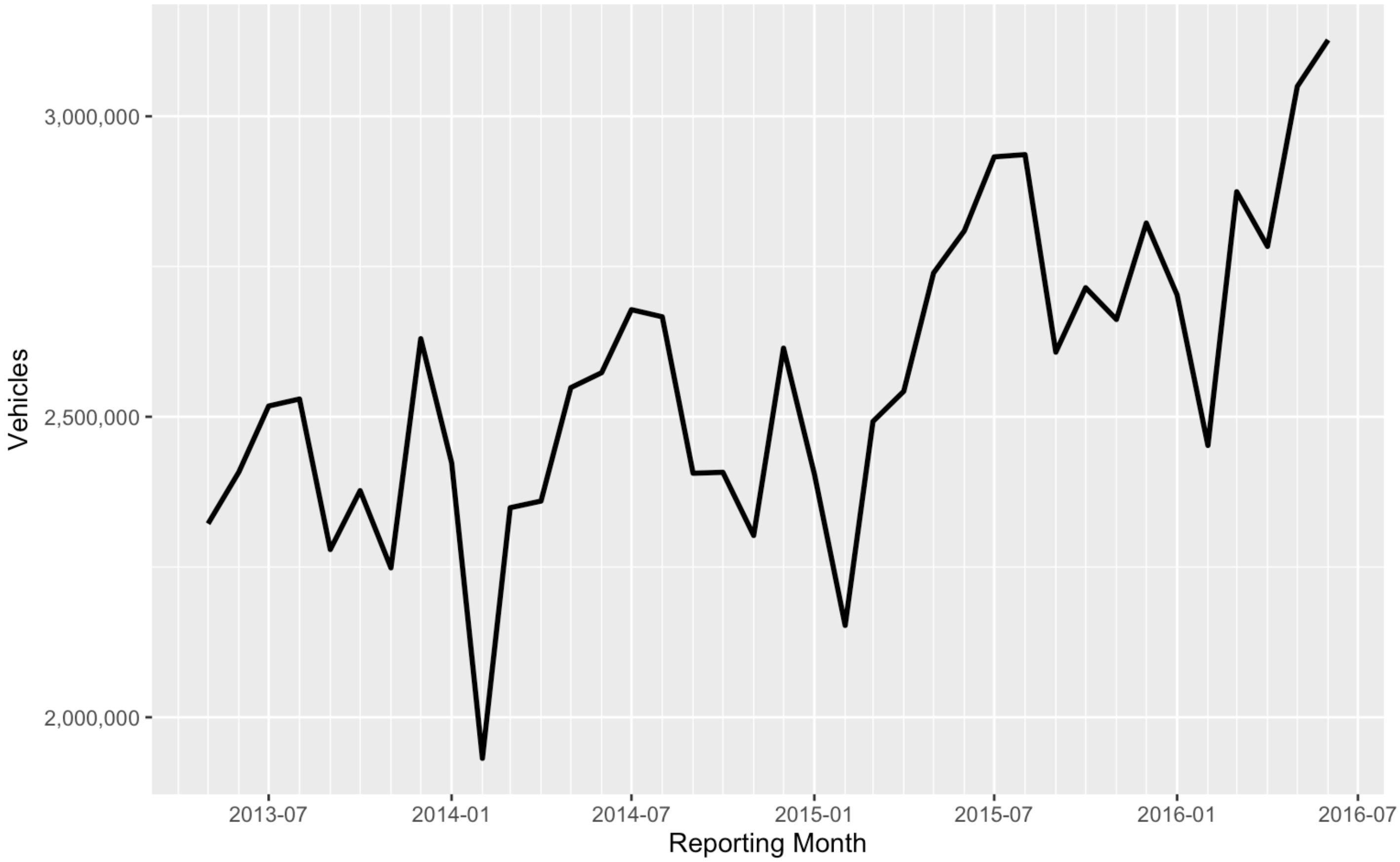
We'll correct for this by editing exits for that month to be the "# of entries in 2014-09" + "average difference between exits and entries".



5.2.3 Part 3

Lastly, exits outpace entries every month. In reality, a vehicle that exits had to have entered at some point, and vice versa.

For our analysis, we only need one number per month, so we'll use the maximum of each months entry or exit.



5.3 Code

```
work_dir="/Users/bradychiu/Dropbox/r/lax_ground_vehicle_traffic_volume"
data_dir="data/"
deliverable_dir="deliverables/"
# setwd(work_dir)

package_loader<-function(package_names){
  for(i in package_names){
    if(!i %in% rownames(installed.packages())) install.packages(i, repos="http://cran.r-
project.org")
    library(i, character.only=T)
  }
}
package_names<-c("data.table", "dplyr", "forecast", "ggplot2", "knitr", "lubridate", "st
ats", "stringr", "tidyr", "tseries")
package_loader(package_names)
```

```
get_data<-function(data_file_paths){
  bind_rows(lapply(data_file_paths, fread, na.strings="", stringsAsFactors=F)) %>%
    data.frame()
}

get_table<-function(df,dig=2){
  kable(df, align='r', digits=dig, format.args=list(big.mark=","))
}

# calculate mean absolute error
get_mae<-function(actuals, predicted){
  err<-abs(actuals-predicted)
  return (mean(err))
}
# calculate mean absolute percentage error
get_mape<-function(actuals, predicted){
  err<-abs(actuals-predicted)/actuals
  return (mean(err))
}
# calculate mean squared error
get_mse<-function(actuals, predicted){
  err<-(actuals-predicted)^2
  return(mean(err))
}
# calculate mean percentage error
get_mpe<-function(actuals, predicted){
  err<-(actuals-predicted)/actuals
  return(mean(err))
}
```

```
validation_periods<-3
```

```
# file.remove(file.path(work_dir, data_dir, "lax_ground_vehicle_traffic_volume.Rds"))
if(!file.exists(file.path(work_dir, data_dir, "lax_ground_vehicle_traffic_volume.Rds")))
{
  file_names<-c(
    "Los_Angeles_International_Airport_-_Ground_Vehicle_Traffic_Volume.csv"
  )
  lax_ground_vehicle_traffic_volume<-get_data(file.path(work_dir, data_dir, file_names))
 %>%
    mutate(ReportingMonth=mdy_hms(ReportingMonth, tz="America/Los_Angeles")) %>%
    dplyr::rename(reporting_month=ReportingMonth,
                  action=EntryExit,
                  level=UpperLower,
                  vehicles=TOTAL.VEHICLES) %>%
    arrange(reporting_month, action, level)
  saveRDS(lax_ground_vehicle_traffic_volume, file.path(work_dir, data_dir, "lax_ground_v
ehicle_traffic_volume.Rds"))
}else{
  lax_ground_vehicle_traffic_volume<-readRDS(file.path(work_dir, data_dir, "lax_ground_v
ehicle_traffic_volume.Rds"))
}
```

```
lax_ground_vehicle_traffic_volume %>%
  group_by(reporting_month, action) %>%
  summarize(vehicles=sum(vehicles)) %>%
  ungroup() %>%
  ggplot(aes(x=reporting_month, y=vehicles, color=action, group=action))+
  geom_line(size=1)+
  scale_x_datetime(name="Reporting Month",
                   date_minor_breaks="1 months")+
  scale_y_continuous(name="Vehicles",
                     breaks=seq(0,10000000,500000),
                     labels=format(seq(0,10000000,500000), big.mark=",", scientific=F))+
  scale_color_discrete(name="Action")+
  theme()
```

```
lax_ground_vehicle_traffic_volume %>%
  filter(year(reporting_month)==2014,
         month(reporting_month)==4) %>%
  get_table()
```

```
lax_ground_vehicle_traffic_volume_edit_1<-lax_ground_vehicle_traffic_volume %>%
  unique()

lax_ground_vehicle_traffic_volume_edit_1 %>%
  filter(year(reporting_month)==2014,
         month(reporting_month)==4) %>%
  get_table()
```

```
mean_exit_entry_delta<-lax_ground_vehicle_traffic_volume_edit_1 %>%
  group_by(reporting_month, action) %>%
  summarize(vehicles=sum(vehicles)) %>%
  ungroup() %>%
  spread(action, vehicles) %>%
  mutate(exit_entry_delta=ifelse(Exit>Entry, Exit-Entry, NA)) %>%
  .$exit_entry_delta %>%
  mean(na.rm=T)

lax_ground_vehicle_traffic_volume_edit_2<-lax_ground_vehicle_traffic_volume_edit_1 %>%
  group_by(reporting_month, action) %>%
  summarize(vehicles=sum(vehicles)) %>%
  ungroup() %>%
  spread(action, vehicles) %>%
  mutate(Exit=ifelse(reporting_month==ymd("2014-09-01", tz="America/Los_Angeles"),
Entry+mean_exit_entry_delta, Exit)) %>%
  gather(action, vehicles, -reporting_month)

lax_ground_vehicle_traffic_volume_edit_2 %>%
  ggplot(aes(x=reporting_month, y=vehicles, color=action, group=action))+
  geom_line(size=1)+
  scale_x_datetime(name="Reporting Month",
                   date_minor_breaks="1 months")+
  scale_y_continuous(name="Vehicles",
                     breaks=seq(0,10000000,500000),
                     labels=format(seq(0,10000000,500000), big.mark=",", scientific=F))+
  scale_color_discrete(name="Action")
```

```
lax_ground_vehicle_traffic_volume_cleaned<-lax_ground_vehicle_traffic_volume_edit_2 %>%
  group_by(reporting_month) %>%
  summarize(vehicles=max(vehicles)) %>%
  ungroup()

lax_ground_vehicle_traffic_volume_cleaned %>%
  ggplot(aes(x=reporting_month, y=vehicles))+
  geom_line(size=1)+
  scale_x_datetime(name="Reporting Month",
                   date_minor_breaks="1 months")+
  scale_y_continuous(name="Vehicles",
                     breaks=seq(0,10000000,500000),
                     labels=format(seq(0,10000000,500000), big.mark=",", scientific=F))
```

```
lax_vehicles<-lax_ground_vehicle_traffic_volume_cleaned %>%
  mutate(dataset=ifelse(row_number()<=nrow(.)-validation_periods, "training", "test"))
lax.training<-lax_vehicles %>%
  filter(dataset=="training") %>%
  dplyr::select(-dataset)
lax.test<-lax_vehicles %>%
  filter(dataset=="test") %>%
  dplyr::select(-dataset)

lax_vehicles %>%
  ggplot(aes(x=reporting_month, y=vehicles, color=dataset))+
  geom_line(size=1)+
  ggtitle("Hold Out Data for Validation Testing")+
  scale_x_datetime(name="Reporting Month")+
  scale_y_continuous(name="Vehicles",
                     breaks=seq(-1000000,1000000,250000),
                     labels=format(seq(-1000000,1000000,250000), big.mark=",", scientifi
c=F))

lax.ts12<-ts(lax.training$vehicles, frequency=12, start=c(year(min(lax.training$reportin
g_month)),
                                                          month(min(lax.training$reporti
ng_month))))
lax.ts12
```

```
lax.training %>%
  mutate(diff1=vehicles-dplyr::lag(vehicles, 1)) %>%
  ggplot(aes(x=reporting_month, y=diff1))+
  geom_line(size=1)+
  ggtitle("LAX Ground Vehicles - I(1)")+
  scale_x_datetime(name="Reporting Month")+
  scale_y_continuous(name="Vehicles",
                     breaks=seq(-1000000,1000000,250000),
                     labels=format(seq(-1000000,1000000,250000), big.mark=",", scientifi
c=F))
```

```
acf(diff(lax.training$vehicles),
     lag.max=sum(!is.na(lax.training$vehicles)))
```

```
# http://www.statosphere.com.au/check-time-series-stationary-r/
Box.test(diff(lax.training$vehicles),
         lag=20,
         type="Ljung-Box")
```

```
adf.test(diff(lax.training$vehicles), alternative="stationary")
```

```
kpss.test(diff(lax.training$vehicles))
```

```
lax.stl<-stl(lax.ts12, s.window="periodic", t.window = 24)
plot(lax.stl)

lax.training %>%
  mutate("Trend"=lax.stl$time.series[,2],
         "Trend + Seasonal"=lax.stl$time.series[,1]+lax.stl$time.series[,2]) %>%
  dplyr::rename(actuals=vehicles) %>%
  gather(metric, value, actuals:`Trend + Seasonal`) %>%
  mutate(metric=factor(metric, levels=c("Trend", "Trend + Seasonal"))) %>%
  filter(metric!="actuals") %>%
  ggplot(aes(x=reporting_month, y=value, color=metric, group=metric))+
  geom_line(size=1)+
  ggtitle("STL Decomposition")+
  scale_x_datetime(name="Reporting Month",
                   date_minor_breaks="2 months")+
  scale_y_continuous(name="Vehicles",
                     breaks=seq(0,10000000,500000),
                     labels=format(seq(0,10000000,500000), big.mark=",", scientific=F))
```

```
lax.training %>%
  mutate("Seasonal + Trend"=lax.stl$time.series[,1]+lax.stl$time.series[,2]) %>%
  dplyr::rename(Actuals=vehicles) %>%
  gather(metric, value, Actuals:`Seasonal + Trend`) %>%
  ggplot(aes(x=reporting_month, y=value, color=metric, group=metric))+
  geom_line(size=1)+
  ggtitle("STL Decomposition vs. Actuals")+
  scale_x_datetime(name="Reporting Month",
                   date_minor_breaks="2 months")+
  scale_y_continuous(name="Vehicles",
                     breaks=seq(0,10000000,500000),
                     labels=format(seq(0,10000000,500000), big.mark=",", scientific=F))

lax.stl.fcast<-data.frame(forecast.stl(lax.stl, h=validation_periods, method="ets"))
lax.stl.fcast %>%
  get_table()

lax.stl.predict<-lax.test %>%
  dplyr::rename(actuals=vehicles) %>%
  mutate(predicted=lax.stl.fcast$Point.Forecast)

lax.training %>%
  dplyr::rename(actuals=vehicles) %>%
  gather(metric, value, -reporting_month) %>%
  bind_rows(lax.stl.predict %>%
              gather(metric, value, -reporting_month)) %>%
  ggplot(aes(x=reporting_month, y=value,
             color=factor(metric,
                          levels=c("predicted", "actuals"))))+
  geom_line(size=1)+
  ggtitle("STL Decomposition Validation")+
  scale_x_datetime(name="Reporting Month",
                   date_minor_breaks="2 months")+
  scale_y_continuous(name="Vehicles",
                     breaks=seq(0,10000000,500000),
                     labels=format(seq(0,10000000,500000), big.mark=",", scientific=F))+
  scale_color_hue(name="")

stl_validation<-data.frame(model="STL",
                           mae=get_mae(lax.stl.predict$actuals, lax.stl.predict$predicte
d),
                           mape=get_mape(lax.stl.predict$actuals, lax.stl.predict$predic
ted),
                           mse=get_mse(lax.stl.predict$actuals, lax.stl.predict$predicte
d),
                           mpe=get_mpe(lax.stl.predict$actuals, lax.stl.predict$predicte
d),
                           stringsAsFactors=F)
stl_validation %>%
  get_table(4)

forecast_validation<-stl_validation
```

```
# lax.hw<-HoltWinters(lax.ts12, gamma=F) # trend only
lax.hw<-HoltWinters(lax.ts12) # seasonal + trend
lax.hw
```

```
lax.hw.fcast<-data.frame(forecast.HoltWinters(lax.hw, h=validation_periods))
lax.hw.fcast %>%
  get_table()

lax.hw.predict<-lax.test %>%
  dplyr::rename(actuals=vehicles) %>%
  mutate(predicted=lax.hw.fcast$Point.Forecast)

lax.training %>%
  dplyr::rename(actuals=vehicles) %>%
  gather(metric, value, -reporting_month) %>%
  bind_rows(lax.hw.predict %>%
              gather(metric, value, -reporting_month)) %>%
  ggplot(aes(x=reporting_month, y=value,
             color=factor(metric,
                          levels=c("predicted", "actuals"))))+
  geom_line(size=1)+
  ggtitle("Holt-Winters Validation")+
  scale_x_datetime(name="Reporting Month",
                   date_minor_breaks="2 months")+
  scale_y_continuous(name="Vehicles",
                     breaks=seq(0,10000000,500000),
                     labels=format(seq(0,10000000,500000), big.mark=",", scientific=F))+
  scale_color_hue(name="")

hw_validation<-data.frame(model="Holt-Winters",
                          mae=get_mae(lax.hw.predict$actuals, lax.hw.predict$predicted),
                          mape=get_mape(lax.hw.predict$actuals,
lax.hw.predict$predicted),
                          mse=get_mse(lax.hw.predict$actuals, lax.hw.predict$predicted),
                          mpe=get_mpe(lax.hw.predict$actuals, lax.hw.predict$predicted))
hw_validation %>%
  get_table(4)

forecast_validation<-forecast_validation %>%
  bind_rows(hw_validation)
```

```
lax.autoarima<-auto.arima(lax.ts12, allowdrift=F)
lax.autoarima
```

```
lax.autoarima.fcast<-data.frame(forecast(lax.autoarima, h=validation_periods))
lax.autoarima.fcast %>%
  get_table()

lax.autoarima.predict<-lax.test %>%
  dplyr::rename(actuals=vehicles) %>%
  mutate(predicted=lax.autoarima.fcast$Point.Forecast)

lax.training %>%
  dplyr::rename(actuals=vehicles) %>%
  gather(metric, value, -reporting_month) %>%
  bind_rows(lax.autoarima.predict %>%
              gather(metric, value, -reporting_month)) %>%
  ggplot(aes(x=reporting_month, y=value,
             color=factor(metric,
                          levels=c("predicted", "actuals"))))+
  geom_line(size=1)+
  ggtitle("Auto-Arima Validation")+
  scale_x_datetime(name="Reporting Month",
                   date_minor_breaks="2 months")+
  scale_y_continuous(name="Vehicles",
                     breaks=seq(0,10000000,500000),
                     labels=format(seq(0,10000000,500000), big.mark=",", scientific=F))+
  scale_color_hue(name="")

arima_validation<-data.frame(model="Arima",
                             mae=get_mae(lax.autoarima.predict$actuals, lax.autoarima.pr
edict$predicted),
                             mape=get_mape(lax.autoarima.predict$actuals, lax.autoarima.
predict$predicted),
                             mse=get_mse(lax.autoarima.predict$actuals, lax.autoarima.pr
edict$predicted),
                             mpe=get_mpe(lax.autoarima.predict$actuals, lax.autoarima.pr
edict$predicted))

arima_validation %>%
  get_table(4)

forecast_validation<-forecast_validation %>%
  bind_rows(arima_validation)
```