Data607-Week05-Assignment Tidying and Transforming Data

Vinayak Kamath

2/29/2020

		Los Angeles	Phoenix	San Diego	San Francisco	Seattle
ALASKA	on time	497	221	212	503	1,841
	delayed	62	12	20	102	305
AM WEST	on time	694	4,840	383	320	201
	delayed	117	415	65	129	61

Source: *Numbersense*, Kaiser Fung, McGraw Hill, 2013

(1) Create a .CSV file (or optionally, a MySQL database!) that includes all of the information above. You're encouraged to use a "wide" structure similar to how the information appears above, so that you can practice tidying and transformations as described below.

CSV File created using Microsft Excel. File can found in Git Repository.

---

(2) Read the information from your .CSV file into R, and use tidyr and dplyr as needed to tidy and transform your data.

```
# Reading the CSV file from GIT repository and loading into dataframe:
theUrl <- "https://raw.githubusercontent.com/kamathvk1982/Data607-Week05/master/ArrivalDelays.csv"
ArrivalDelays.df <- read.csv(file = theUrl, header = T , sep = ','
                             , na.strings=c("NA","NaN", "") )
kable(ArrivalDelays.df)
```

X	X.1	Los.Angeles	Pheonix	San.Diego	San.Francisco	Seattle
Alaska	on time	497	221	212	503	1,841
NA	delayed	62	12	20	102	305
NA	NA	NA	NA	NA	NA	NA
AM WEST	on time	694	4,840	383	320	201
NA	delayed	117	415	65	129	61

```
# Renaming the first two columns to meaningfull names:
colnames(ArrivalDelays.df)[colnames(ArrivalDelays.df)
                           %in% c("X", "X.1")] <- c("Airline", "Status")

# Use the tidyr->drop_na function to drop the row on column Status having NA value
ArrivalDelays.df <- drop_na(ArrivalDelays.df, Status)
kable(ArrivalDelays.df)
```

	Airline	Status	Los.Angeles	Pheonix	San.Diego	San.Francisco	Seattle
1	Alaska	on time	497	221	212	503	1,841
2	NA	delayed	62	12	20	102	305
4	AM WEST	on time	694	4,840	383	320	201
5	NA	delayed	117	415	65	129	61

```
# Use the tidyr->fill function to get Airline name on the row for the delayed data:
# fill() function, by defualt downward direction, helps to  fill NA value from recent
# non-NA values
ArrivalDelays.df <- fill(ArrivalDelays.df, Airline)
kable(ArrivalDelays.df)
```

Airline	Status	Los.Angeles	Pheonix	San.Diego	San.Francisco	Seattle
Alaska	on time	497	221	212	503	1,841
Alaska	delayed	62	12	20	102	305
AM WEST	on time	694	4,840	383	320	201
AM WEST	delayed	117	415	65	129	61

```
# Above three steps have helped us to get the data in a format that helps to work
# with the final set of data and to work with it in a better way now.

# Next, we will tidy the data by reshaping the data layput in the table by using
# tidyr->gather function to move the airport column names into a key column,
```

```
# gathering the column values into a single value column.

# We will create two dataset for Ontime and Delayed information and lter merge them
# to form a single final dataset

OnTime.ArrivalDelays.df <- ArrivalDelays.df %>%
  filter(Status == 'on time')  %>%
  gather( `Los.Angeles`,`Pheonix`,`San.Diego`,
          `San.Francisco`,`Seattle`,key = "Airport", value = "On.Time") %>%
  select( Airline, Airport, On.Time) %>%
  arrange(Airline, Airport)

kable(OnTime.ArrivalDelays.df)
```

Airline	Airport	On.Time
Alaska	Los.Angeles	497
Alaska	Pheonix	221
Alaska	San.Diego	212
Alaska	San.Francisco	503
Alaska	Seattle	1,841
AM WEST	Los.Angeles	694
AM WEST	Pheonix	4,840
AM WEST	San.Diego	383
AM WEST	San.Francisco	320
AM WEST	Seattle	201

```
delayed.ArrivalDelays.df <- ArrivalDelays.df %>%
  filter(Status == 'delayed')  %>%
  gather( `Los.Angeles`,`Pheonix`,`San.Diego`,
          `San.Francisco`,`Seattle`,key = "Airport", value = "Delayed") %>%
  select( Airline, Airport, Delayed) %>%
  arrange(Airline, Airport)

kable(delayed.ArrivalDelays.df)
```

Airline	Airport	Delayed
Alaska	Los.Angeles	62
Alaska	Pheonix	12
Alaska	San.Diego	20
Alaska	San.Francisco	102
Alaska	Seattle	305
AM WEST	Los.Angeles	117
AM WEST	Pheonix	415
AM WEST	San.Diego	65
AM WEST	San.Francisco	129
AM WEST	Seattle	61

```
# We will now merge the above two dataset to one fianl dataset based on common keys
# Airline and Airport; we will use dplyr->full_join
```

```r
final.ArrivalDelays.df <- full_join(OnTime.ArrivalDelays.df, delayed.ArrivalDelays.df
                                    ,by =  c("Airline", "Airport"), copy=FALSE)

# We will add / dplyr->mutate a new column to get the percent of dealyed against total flights
# an airline does at an airport

final.ArrivalDelays.df$On.Time  <- as.double(sub(',','',final.ArrivalDelays.df$On.Time))
final.ArrivalDelays.df$Delayed  <- as.double(sub(',','',final.ArrivalDelays.df$Delayed))

final.ArrivalDelays.df <- mutate(final.ArrivalDelays.df, Percent.Delayed = round((Delayed/
                                    (On.Time+Delayed) )*100, 2) )

# Final Transformed Data ; Ready for further analysis!:
kable(final.ArrivalDelays.df)
```

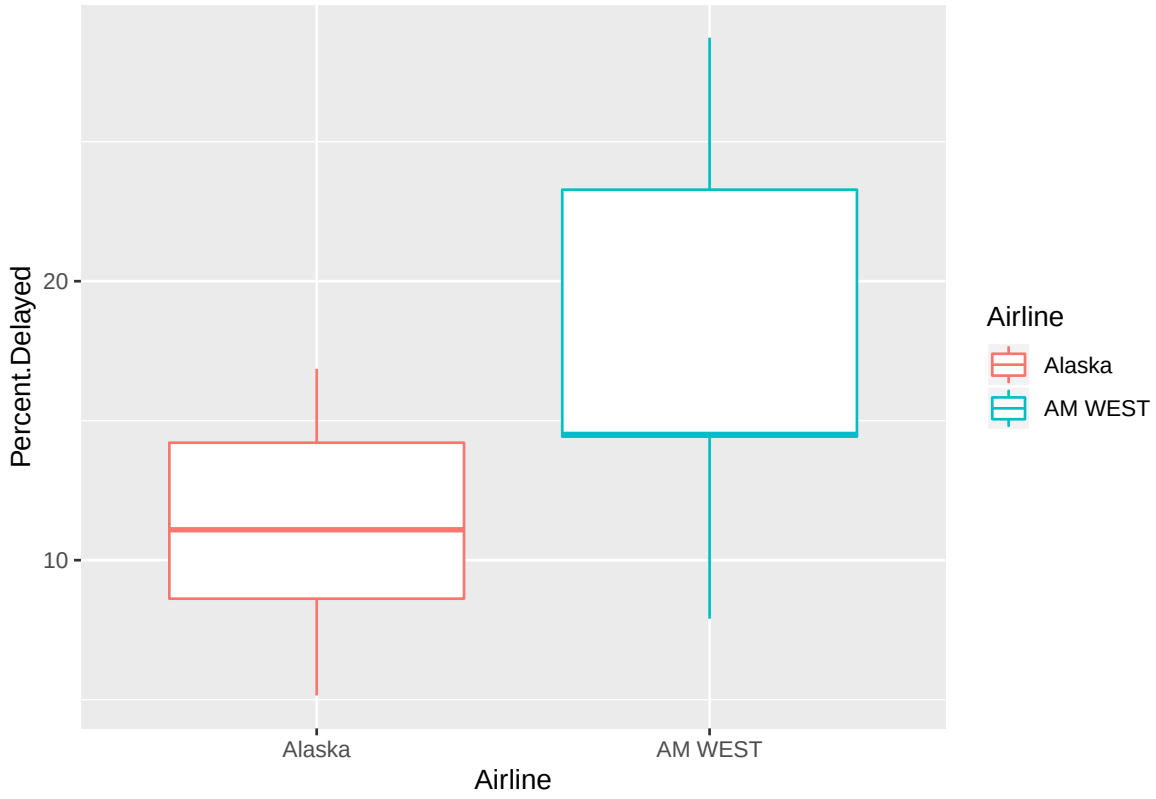
Airline	Airport	On.Time	Delayed	Percent.Delayed
Alaska	Los.Angeles	497	62	11.09
Alaska	Pheonix	221	12	5.15
Alaska	San.Diego	212	20	8.62
Alaska	San.Francisco	503	102	16.86
Alaska	Seattle	1841	305	14.21
AM WEST	Los.Angeles	694	117	14.43
AM WEST	Pheonix	4840	415	7.90
AM WEST	San.Diego	383	65	14.51
AM WEST	San.Francisco	320	129	28.73
AM WEST	Seattle	201	61	23.28

(3) Perform analysis to compare the arrival delays for the two airlines.

```
# 1. We can get the Mean and Standard Deviation for each airline:
final.ArrivalDelays.df %>%
  group_by(Airline) %>%
    summarise(Mean=mean(Percent.Delayed), SD=sd(Percent.Delayed))
```

```
## # A tibble: 2 x 3
##   Airline  Mean    SD
##   <fct>   <dbl> <dbl>
## 1 Alaska   11.2  4.59
## 2 AM WEST  17.8  8.21
```

```
# Plotting the boxplot to check on the outliers:
ggplot(final.ArrivalDelays.df, aes(x=Airline, y=Percent.Delayed , color=Airline)) +
  geom_boxplot()
```



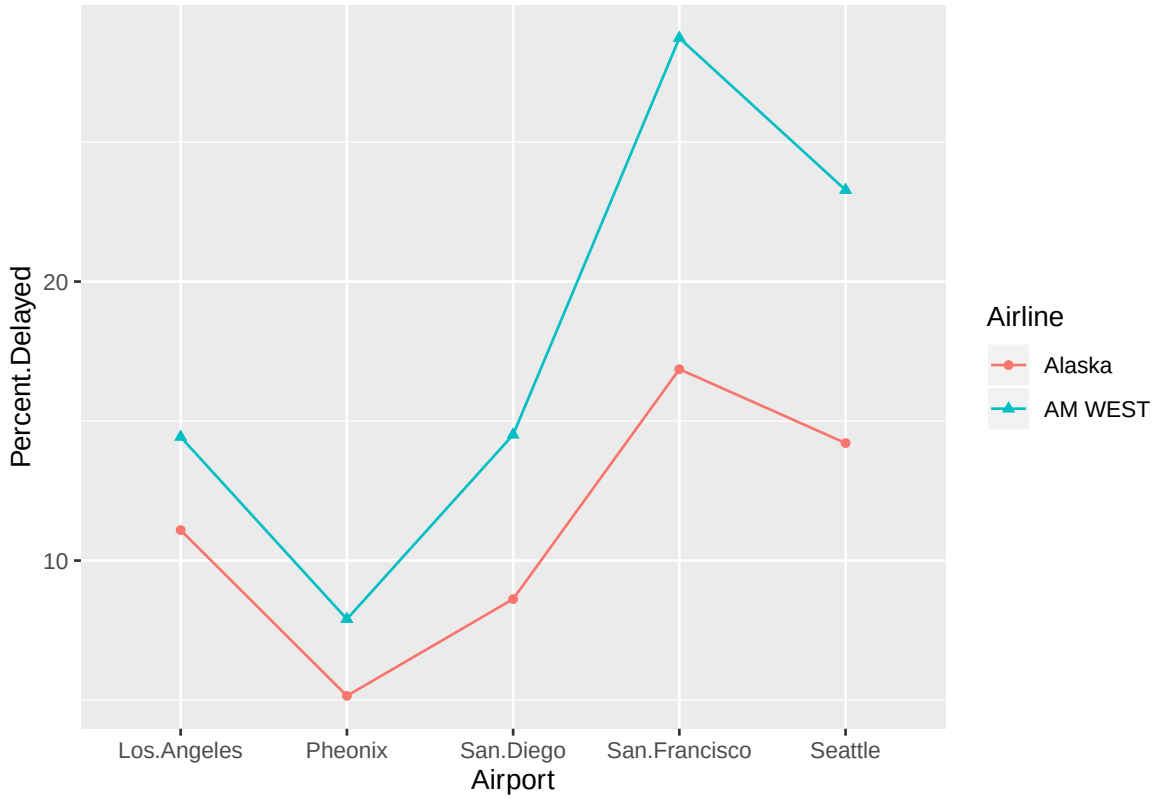
-> Based on above, we can say AM WEST on average is more delayed then Alaska Airlines.

```
# 2. Airport wise we can check the Top Delayed Airline:
final.ArrivalDelays.df %>%
  group_by(Airport) %>%
  top_n(1, Percent.Delayed)
```

```
## # A tibble: 5 x 5
## # Groups:   Airport [5]
```

```
##   Airline Airport       On.Time Delayed Percent.Delayed
##   <fct>   <chr>           <dbl>   <dbl>           <dbl>
## 1 AM WEST Los.Angeles       694     117            14.4
## 2 AM WEST Pheonix          4840     415             7.9
## 3 AM WEST San.Diego         383      65            14.5
## 4 AM WEST San.Francisco     320     129            28.7
## 5 AM WEST Seattle           201      61            23.3
```

```r
# Plotting the line plot to check on same:
ggplot(final.ArrivalDelays.df, aes(x = Airport, y = Percent.Delayed, group = Airline,color
                                   = Airline,shape=Airline) ,xlab = "Airport" , ylab = "Percent.Del
  geom_line() + geom_point((aes(shape=Airline)))
```



-> Based on above, we can say AM WEST had highest delayed for each airport as well.
