



# *Data Transformation III*

## Data Transformation III Info



- Finish Reading Chapter 3 and Practice the Code in R4DS
- Covers
  - The Pipe
  - Statistical Summaries
  - Grouped Summaries
  - Helpful Functions
- Builds Off Tutorial 3

## The Pipe



- Useful for Combining Multiple Steps of Operations
- Represented by `%>%`
- Reads as “Then”
- Works Like a Composite Function From Algebra

$$\begin{aligned}f(x) &= 3x + 4 \\g(x) &= 2x \\h &= 1\end{aligned}$$



$$\begin{aligned}\text{OUT} &= h \%>\% \\&\quad g() \%>\% \\&\quad \quad f()\end{aligned}$$

$$f(g(h)) = 3(2(1)) + 4 = 10$$

$$\text{OUT} = 10$$

# The Pipe



- Chaining with the Pipe

```
library(tidyverse)

f.pipedream =
  # Acknowledge the Original Data
  flights %>%

  # Input Original Data and Perform Mutations
  mutate(dep_hr=dep_time%%100+(dep_time%%100)/60,
          sched_dep_hr=sched_dep_time%%100+(sched_dep_time%%100)/60,
          arr_hr=arr_time%%100+(arr_time%%100)/60,
          sched_arr_hr=sched_arr_time%%100+(sched_arr_time%%100)/60,
          dep_delay_hr=dep_hr-sched_dep_hr,
          arr_delay_hr=arr_hr-sched_arr_hr,
          gain_hr=arr_delay_hr-dep_delay_hr,
          percent_gain_hr=percent_rank(gain_hr)) %>%

  #Input Modified Data and Select the Variables of Interest
  select(carrier,origin:distance,dep_delay_hr:percent_gain_hr) %>%

  #Input Modified Data and Sort According to Empirical %-iles
  arrange(desc(percent_gain_hr))
```

carrier	origin	dest	air_time	distance	dep_delay_hr	arr_delay_hr	gain_hr	percent_gain_hr
B6	JFK	BQN	NA	1576	-23.90000	3.333333	27.23333	1.0000000
B6	JFK	PSE	NA	1617	-23.65000	3.550000	27.20000	0.9999970
B6	JFK	PSE	NA	1617	-23.80000	2.950000	26.75000	0.9999939
B6	JFK	SJU	NA	1598	-23.58333	3.116667	26.70000	0.9999909
B6	JFK	PSE	NA	1617	-23.76667	2.483333	26.25000	0.9999878



HTML Table: [knitr](#) and [kableExtra](#)

# The Pipe



- Chaining with the Pipe

```
```{r}
f.pipedream2 =
  # Acknowledge the Original Data
  flights %>%

  # Input Original Data and Perform Mutations
  mutate(dep_hr=dep_time%%100+(dep_time%%100)/60,
          sched_dep_hr=sched_dep_time%%100+(sched_dep_time%%100)/60,
          arr_hr=arr_time%%100+(arr_time%%100)/60,
          sched_arr_hr=sched_arr_time%%100+(sched_arr_time%%100)/60,
          dep_delay_hr=dep_hr-sched_dep_hr,
          arr_delay_hr=arr_hr-sched_arr_hr,
          gain_hr=arr_delay_hr-dep_delay_hr,
          percent_gain_hr=percent_rank(gain_hr)) %>%

  #Input Modified Data and select the Variables of Interest
  select(carrier,origin:distance,dep_delay_hr:percent_gain_hr) %>%

  #Input Modified Data and Sort According to Empirical %-iles
  arrange(desc(percent_gain_hr)) %>%

  #Input Modified Data and Remove Flights Missing Air Time
  filter(!is.na(air_time))
```
```

| carrier | origin | dest | air_time | distance | dep_delay_hr | arr_delay_hr | gain_hr  | percent_gain_hr |
|---------|--------|------|----------|----------|--------------|--------------|----------|-----------------|
| B6      | JFK    | PSE  | 214      | 1617     | -23.66667    | 1.133333     | 24.80000 | 0.9999848       |
| B6      | JFK    | PSE  | 214      | 1617     | -23.26667    | 1.500000     | 24.76667 | 0.9999817       |
| B6      | JFK    | BQN  | 199      | 1576     | -21.66667    | 3.050000     | 24.71667 | 0.9999787       |
| B6      | JFK    | LAX  | 317      | 2475     | -22.63333    | 2.050000     | 24.68333 | 0.9999726       |
| B6      | JFK    | PSE  | 200      | 1617     | -23.61667    | 1.066667     | 24.68333 | 0.9999726       |

# The Pipe



- Chaining with the Pipe



```
```{r}
f.pipedream3 =

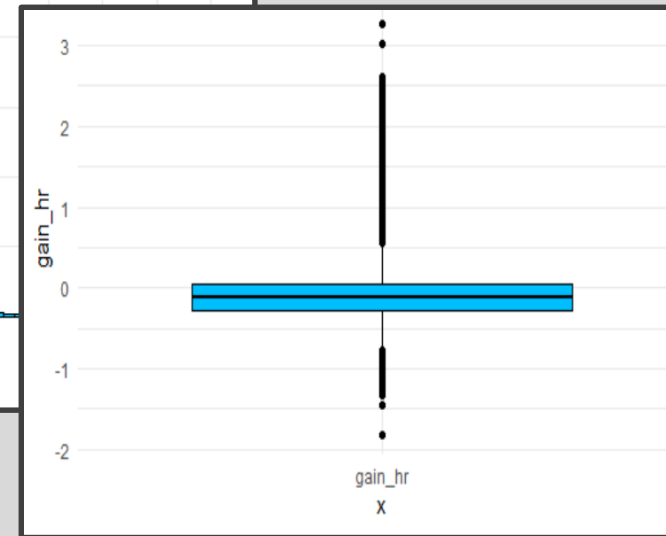
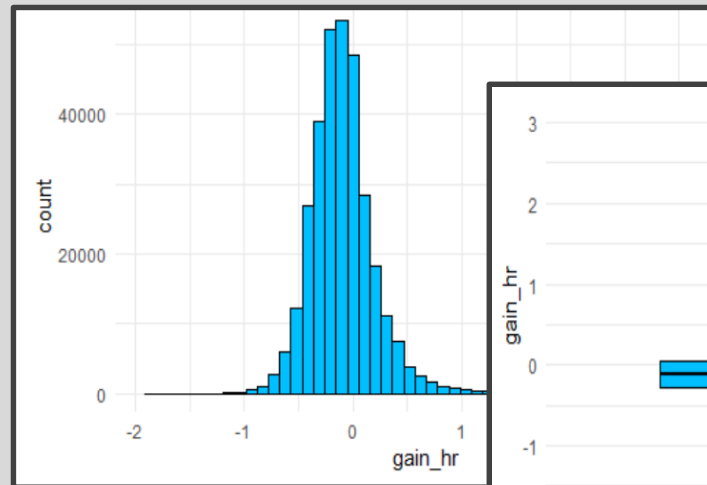
  # Acknowledge the Modified Data
  f.pipedream2 %>%

  # Filter Based on Gain Variable
  filter(abs(gain_hr)<10)
```
```

```
summarize()
```



- Summarizing All Data
- Using Graphics



Both the histogram and the boxplot are made from summary statistics.

**(Statistical Transformations in Ch. 3)**

summarize()



- Summarizing All Data
- Using Tables

```
```{r}
gain_hr.summary1 = summarize(f.pipedream3,
                             n=n(),
                             mean=mean(gain_hr, na.rm=T),
                             var=var(gain_hr, na.rm=T),
                             sd=sd(gain_hr, na.rm=T))

gain_hr.summary2 =
  f.pipedream3 %>%
  summarize(n=n(),
            min=min(gain_hr),
            Q1=quantile(gain_hr, 0.25),
            Q2=quantile(gain_hr, 0.5),
            Q3=quantile(gain_hr, 0.75),
            max=max(gain_hr),
            IQR=Q3-Q1)
```
```

| n      | mean       | var       | sd        |
|--------|------------|-----------|-----------|
| 319966 | -0.0964329 | 0.0879657 | 0.2965902 |

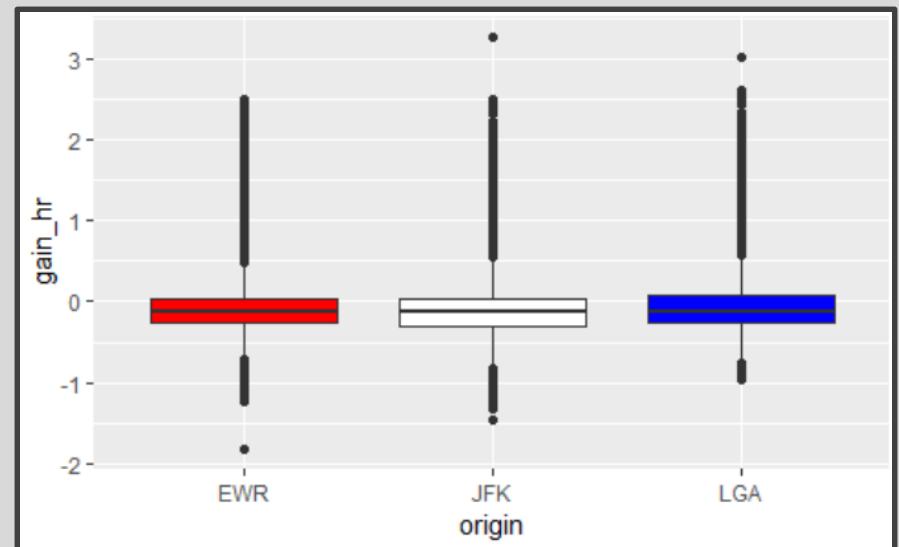
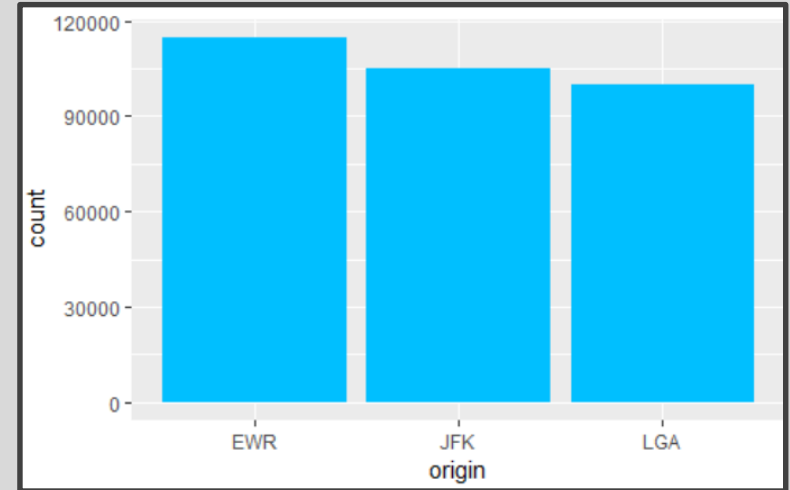
| n      | min       | Q1         | Q2         | Q3   | max      | IQR       |
|--------|-----------|------------|------------|------|----------|-----------|
| 319966 | -1.816667 | -0.2833333 | -0.1166667 | 0.05 | 3.266667 | 0.3333333 |



```
summarize()  
  with  
  group_by()
```



- Summarizing Data by Groups
  - Using Graphics



summarize()  
with  
group\_by()



- Summarizing Data by Groups
- Using Tables

```
```{r}
group.summary1 = f.pipedream3 %>%
  group_by(origin) %>%
  summarize(count=n())

group.summary2 =
  f.pipedream3 %>%
  group_by(origin) %>%
  summarize(
    n=n(),
    min=min(gain_hr),
    Q1=quantile(gain_hr,0.25),
    Q2=quantile(gain_hr,0.5),
    Q3=quantile(gain_hr,0.75),
    max=max(gain_hr),
    IQR=Q3-Q1,
    nLow=sum(gain_hr<Q1-1.5*IQR),
    propHigh=mean(gain_hr>Q3+1.5*IQR)
  ) %>%
  select(-IQR)
```
```

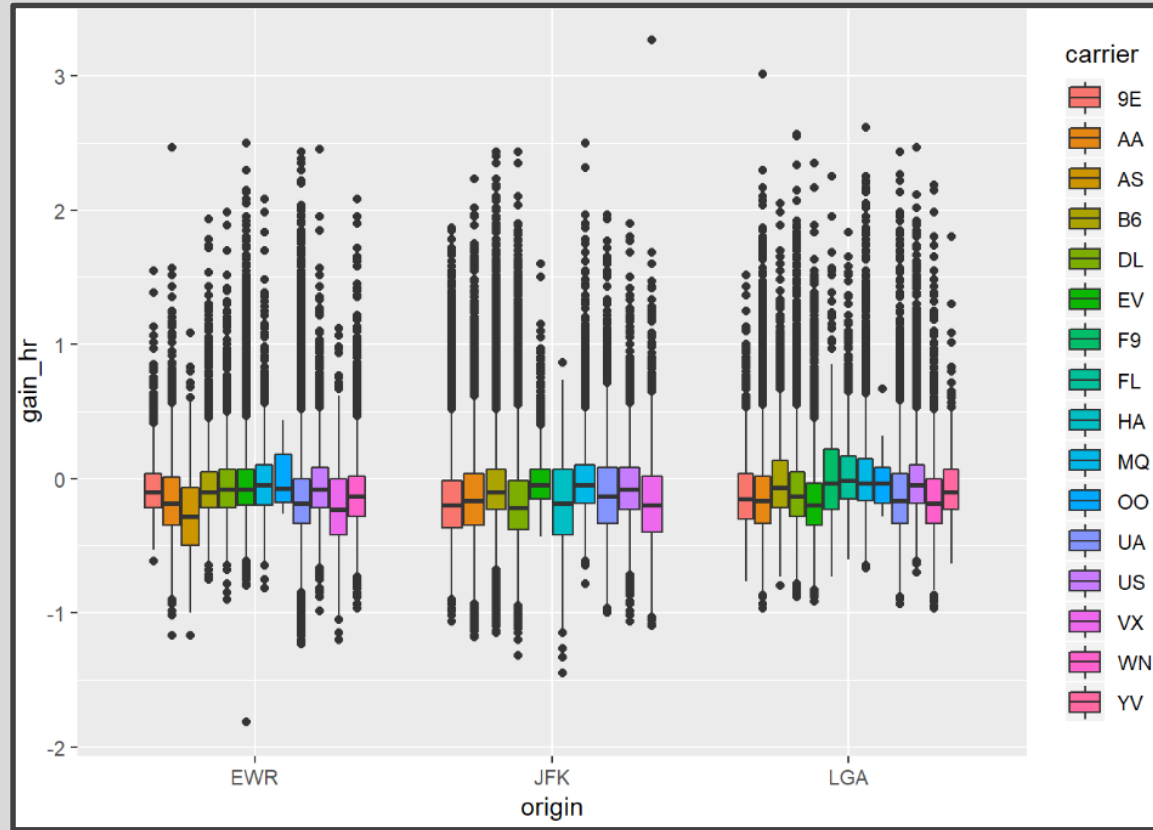
| origin | count  |
|--------|--------|
| EWR    | 114682 |
| JFK    | 105243 |
| LGA    | 100041 |

| origin | n      | min        | Q1         | Q2         | Q3        | max      | nLow | propHigh  |
|--------|--------|------------|------------|------------|-----------|----------|------|-----------|
| EWR    | 114682 | -1.8166667 | -0.2666667 | -0.1166667 | 0.0333333 | 2.500000 | 953  | 0.0294815 |
| JFK    | 105243 | -1.4500000 | -0.3000000 | -0.1333333 | 0.0333333 | 3.266667 | 710  | 0.0314510 |
| LGA    | 100041 | -0.9666667 | -0.2666667 | -0.1166667 | 0.0666667 | 3.016667 | 133  | 0.0277886 |

```
summarize()  
  with  
  group_by()
```



- Multiple Groups
  - Using Graphics



summarize()  
with  
group\_by()



- Multiple Groups
- Using Tables

| origin | carrier | n     | min        | Q1         | Q2         | Q3         | max       |
|--------|---------|-------|------------|------------|------------|------------|-----------|
| EWR    | 9E      | 1193  | -0.6166667 | -0.2166667 | -0.1000000 | 0.0333333  | 1.5500000 |
| EWR    | AA      | 3326  | -1.1666667 | -0.3500000 | -0.1833333 | 0.0125000  | 2.4666667 |
| EWR    | AS      | 704   | -1.1666667 | -0.5000000 | -0.2833333 | -0.0666667 | 1.0833333 |
| EWR    | B6      | 6275  | -0.7500000 | -0.2166667 | -0.1000000 | 0.0500000  | 1.9333333 |
| EWR    | DL      | 4266  | -0.9000000 | -0.2166667 | -0.0833333 | 0.0666667  | 1.9833333 |
| EWR    | EV      | 40571 | -1.8166667 | -0.2000000 | -0.0833333 | 0.0666667  | 2.5000000 |
| EWR    | MQ      | 2086  | -0.8166667 | -0.2000000 | -0.0500000 | 0.1000000  | 2.0833333 |
| EWR    | OO      | 6     | -0.2666667 | -0.1791667 | -0.0750000 | 0.1791667  | 0.4333333 |
| EWR    | UA      | 44390 | -1.2333333 | -0.3333333 | -0.1833333 | 0.0000000  | 2.4333333 |
| EWR    | US      | 4322  | -0.9833333 | -0.2166667 | -0.0833333 | 0.0833333  | 2.4500000 |
| EWR    | VX      | 1521  | -1.2000000 | -0.4166667 | -0.2333333 | 0.0000000  | 1.1166667 |
| EWR    | WN      | 6022  | -0.9666667 | -0.2833333 | -0.1333333 | 0.0166667  | 2.0833333 |
| JFK    | 9E      | 13548 | -1.0666667 | -0.3666667 | -0.2000000 | -0.0166667 | 1.8666667 |
| JFK    | AA      | 13429 | -1.1833333 | -0.3500000 | -0.1666667 | 0.0333333  | 2.2333333 |
| JFK    | B6      | 38920 | -1.1500000 | -0.2333333 | -0.1000000 | 0.0666667  | 2.4333333 |
| JFK    | DL      | 20136 | -1.3166667 | -0.3833333 | -0.2166667 | -0.0166667 | 2.4333333 |
| JFK    | EV      | 1317  | -0.4333333 | -0.1500000 | -0.0500000 | 0.0666667  | 1.6000000 |

## Useful Summary Functions



- Measures of Center
  - `mean()`
  - `median()`
  - `mode()`
- Measures of Spread
  - `var()`
  - `sd()`
  - `IQR()`
  - `mad()`
- Measures of Rank
  - `min()`
  - `max()`
  - `quantile()`

## Useful Summary Functions



- Measures of Position
  - Order Matters
  - `first() = x[1]`
  - `last() = x[length(x)]`
  - `nth(,k) = x[k]`
- Counts
  - `n()`
  - `n_distinct()`
- Counts/Proportions for Logical
  - `sum()`
  - `mean()`
  - Example
    - `sum(x>10)`
    - `mean(x>10)`

Closing



Disperse  
and Make  
Reasonable  
Decisions