Garf’s Analysis

In this document I will monitor and analyze data collected at the end of each shift at Garf’s beginning June 8, 2017.

My goals for this analysis include the following:

- Monitor my income
- Analyze trends to help plan future scheduling
- Monitor and analyze my performance as I strive for continual improvement

There are three main variables of interest: how much I made, how much I made as a percentage of sales, and sales. I will look at each in turn, beginning with **how much I made**.

A Look at How Much I Made

```
summary(garfs$Made)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##     6.0   142.0   225.0   241.4   326.0   768.0 
```

```
garfs %>%
    group_by(year, month) %>%
    summarise(sum(Made), median(Made), mean(Made), n()) %>%
    arrange(desc(year), desc(month))
```

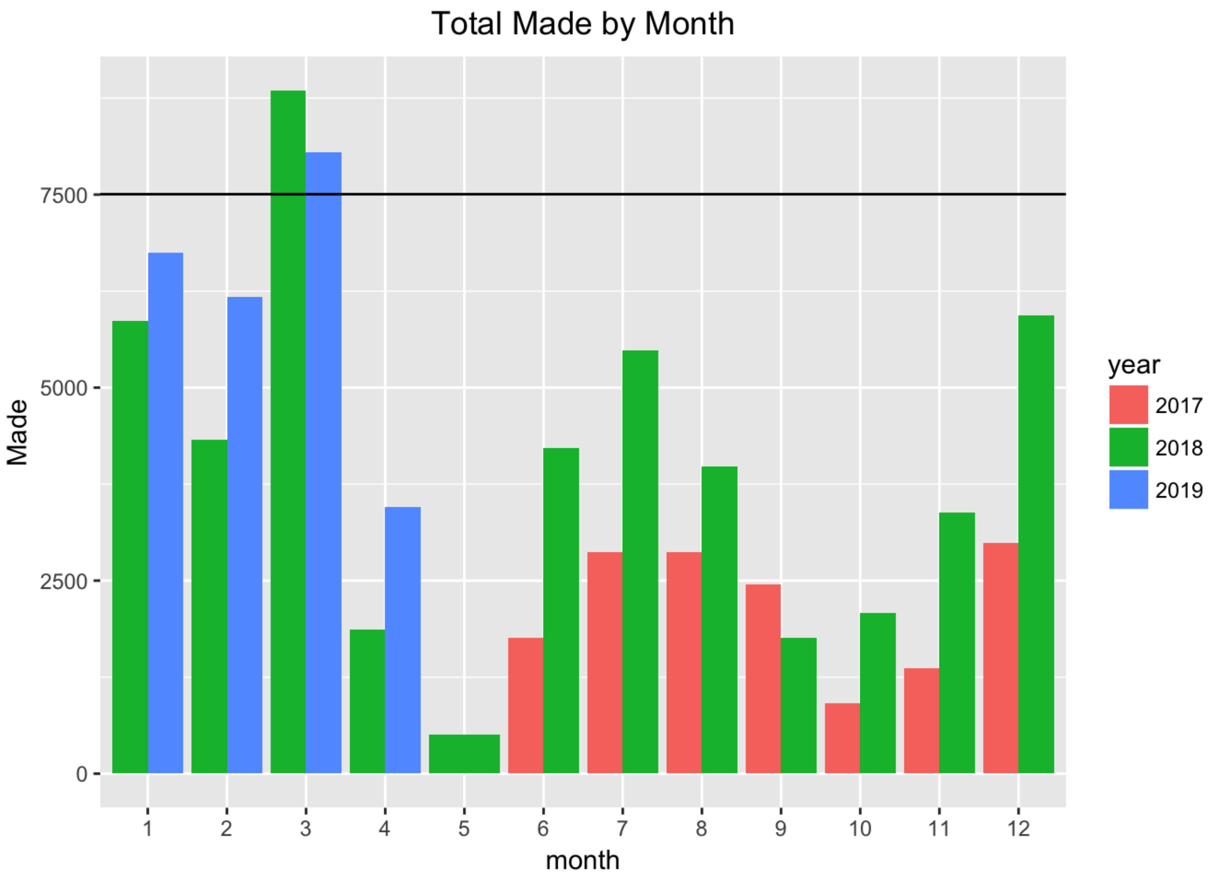
year	month	sum(Made)	median(Made)	mean(Made)	n()
<fctr>	<fctr>	<int>	<dbl>	<dbl>	<int>
2019	4	3452	253.0	265.5385	13
2019	3	8051	345.5	365.9545	22
2019	2	6178	315.5	343.2222	18
2019	1	6742	290.0	293.1304	23
2018	12	5932	290.0	296.6000	20
2018	11	3381	164.5	187.8333	18
2018	10	2076	138.0	129.7500	16
2018	9	1755	174.0	159.5455	11
2018	8	3981	172.5	199.0500	20
2018	7	5487	335.0	322.7647	17

1-10 of 23 rows Previous **1** 2 3 Next

```
made_by_month <- garfs %>%
    group_by(month) %>%
    summarise(made = sum(Made))

mean_made_by_month <- made_by_month %>%
    filter(month != month(Sys.Date())) %>%
        summarise(mean(made))

garfs %>%
    ggplot(aes(month, Made, fill = year)) +
    stat_summary(fun.y = sum, geom = "bar", position = "dodge") +
    geom_hline(aes(yintercept = mean_made_by_month)) +
    ggtitle("Total Made by Month")
```



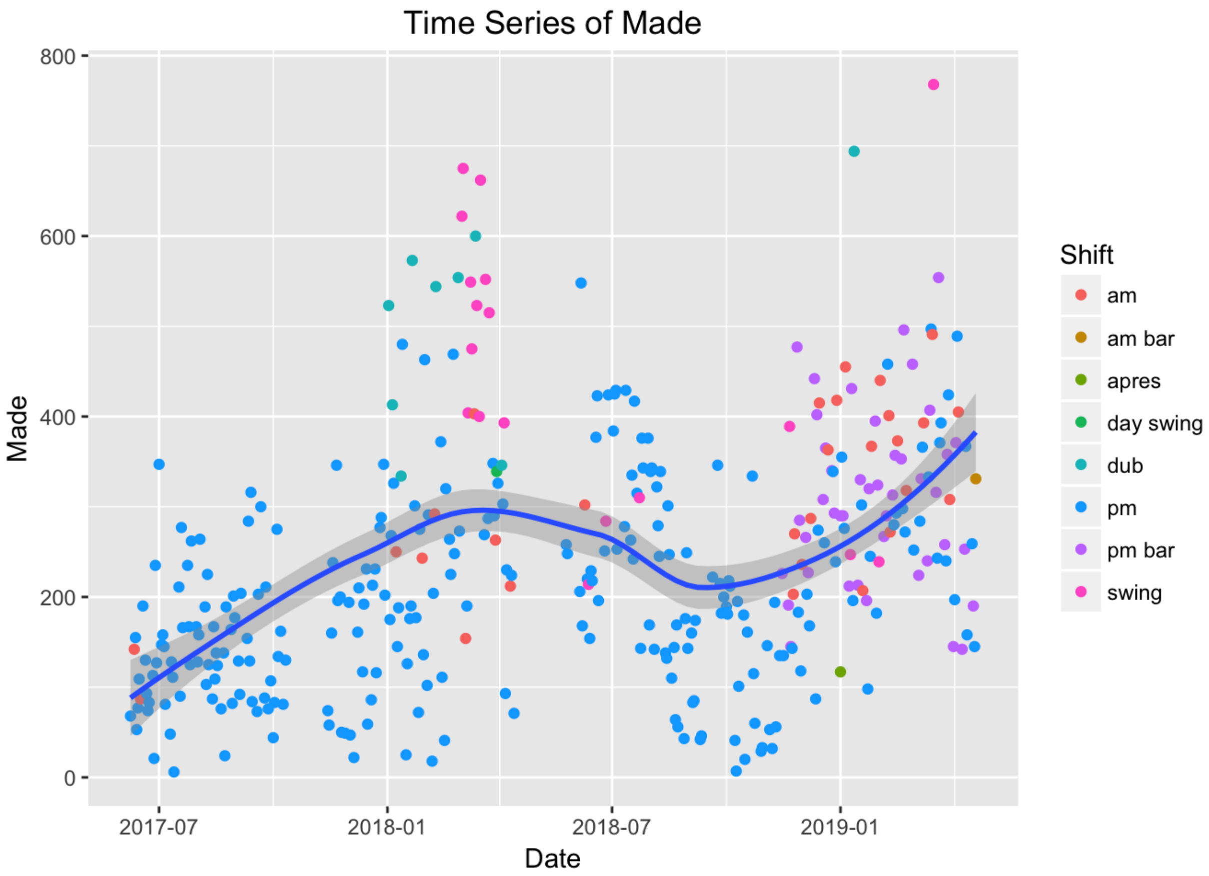
```
garfs %>%
    group_by(year) %>%
    summarise(sum(Made), median(Made), mean(Made), n())
```

year	sum(Made)	median(Made)	mean(Made)	n()
<fctr>	<int>	<dbl>	<dbl>	<int>
2017	15219	136.0	149.2059	102
2018	48237	247.5	259.3387	186
2019	24423	310.5	321.3553	76

3 rows

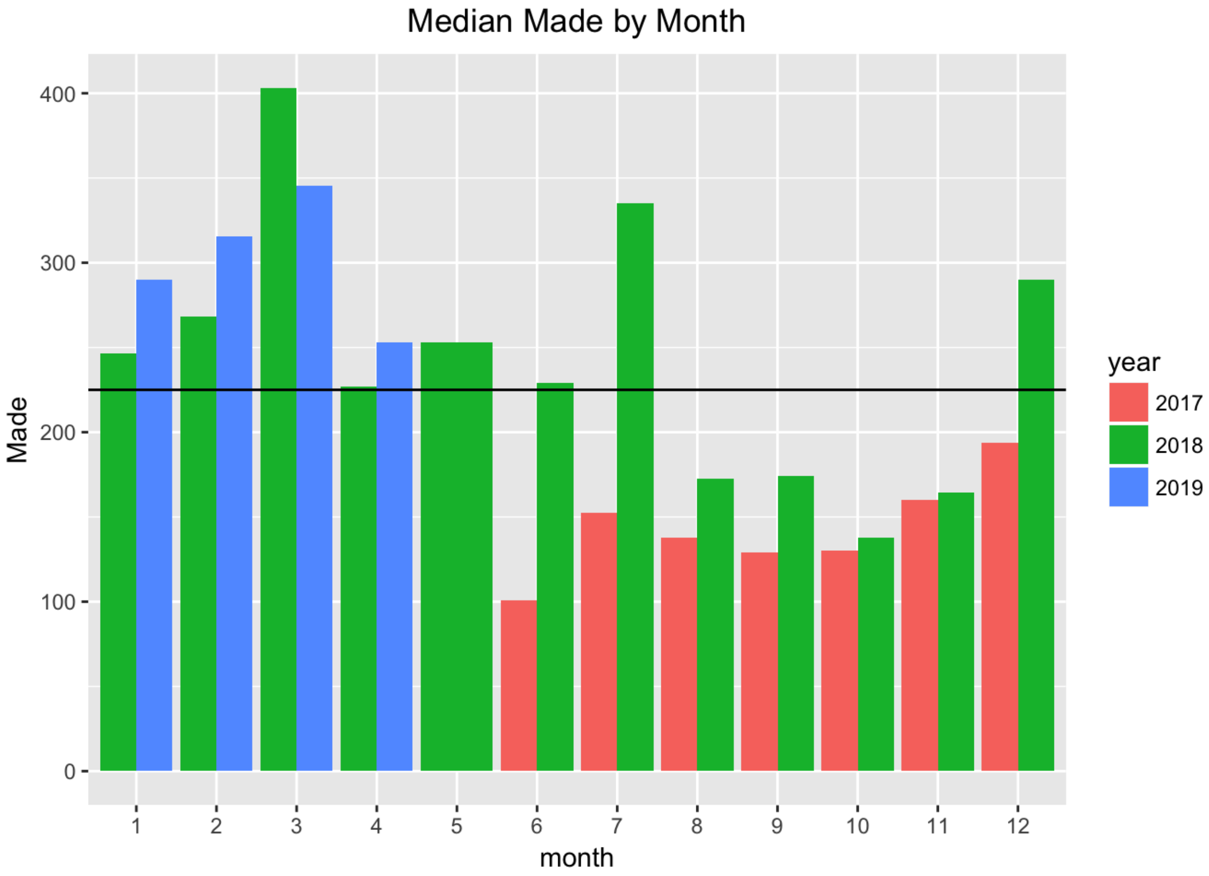
```
garfs %>%
    ggplot(aes(Date, Made)) +
    geom_point(aes(color = Shift)) +
    geom_smooth() +
    ggtitle("Time Series of Made")
```

```
## `geom_smooth()` using method = 'loess'
```



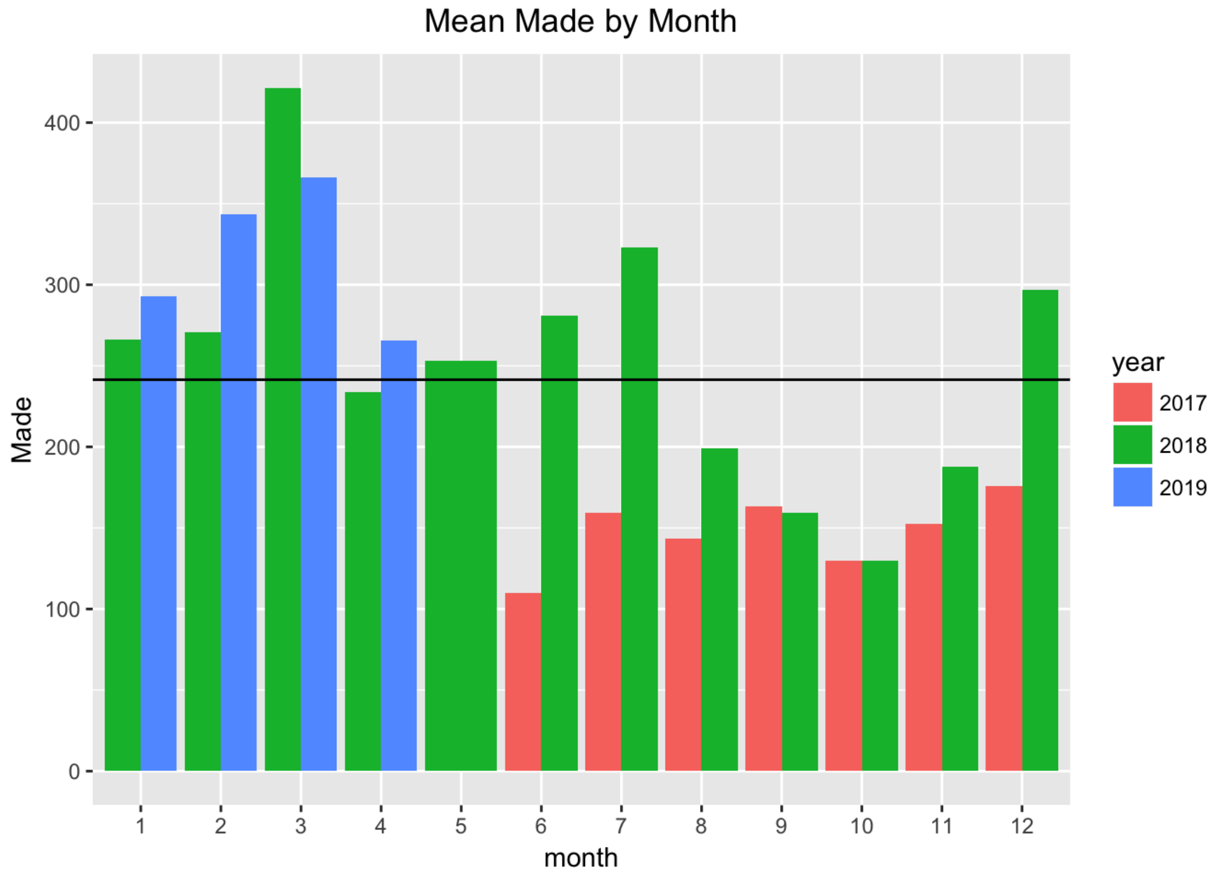
An outlier robust look at how much I made per shift, by month:

```
garfs %>%
    ggplot(aes(month, Made, fill = year)) +
    stat_summary(fun.y = median, geom = "bar", position = "dodge") +
    geom_hline(aes(yintercept = median(Made))) +
    ggtitle("Median Made by Month")
```



A look at how much I made per shift including outliers in order to capture the full variation, by month:

```
garfs %>%
    ggplot(aes(month, Made, fill = year)) +
    stat_summary(fun.y = mean, geom = "bar", position = "dodge") +
    geom_hline(aes(yintercept = mean(Made))) +
    ggtitle("Mean Made by Month")
```



Monthly median and mean of total made:

```
garfs %>%
    group_by(month) %>%
    summarise(tot = sum(Made)) %>%
    summarise(median(tot), mean(tot))
```

median(tot)	mean(tot)
<dbl>	<dbl>
6410	7323.25

1 row

```
# same but excluding May
garfs %>%
    filter(month != 5) %>%
    group_by(month) %>%
    summarise(tot = sum(Made)) %>%
    summarise(median(tot), mean(tot))
```

median(tot)	mean(tot)
<int>	<dbl>
6849	7943

1 row

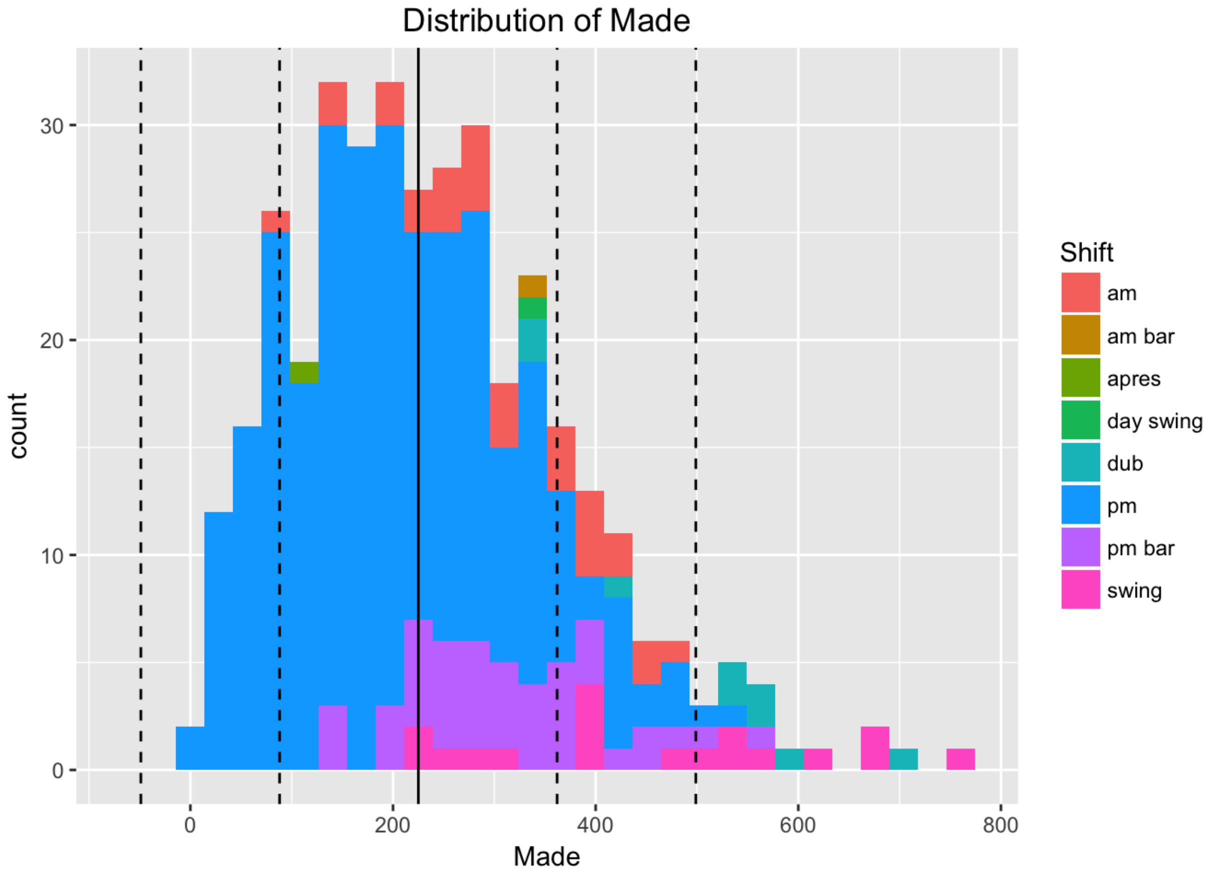
```
# same but excluding May, October, November
garfs %>%
    filter(month != 5, month != 10, month != 11) %>%
    group_by(month) %>%
    summarise(tot = sum(Made)) %>%
    summarise(median(tot), mean(tot))
```

median(tot) <int>	mean(tot) <dbl>
8358	8848.444

1 row

```
garfs %>%
    ggplot(aes(Made)) +
    geom_histogram(aes(fill = Shift)) +
    geom_vline(aes(xintercept = median(Made))) +
    geom_vline(aes(xintercept = median(Made) + sd(Made)), linetype = 2) +
    geom_vline(aes(xintercept = median(Made) - sd(Made)), linetype = 2) +
    geom_vline(aes(xintercept = median(Made) + (2 * sd(Made))), linetype = 2) +
    geom_vline(aes(xintercept = median(Made) - (2* sd(Made))), linetype = 2) +
    ggtitle("Distribution of Made")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



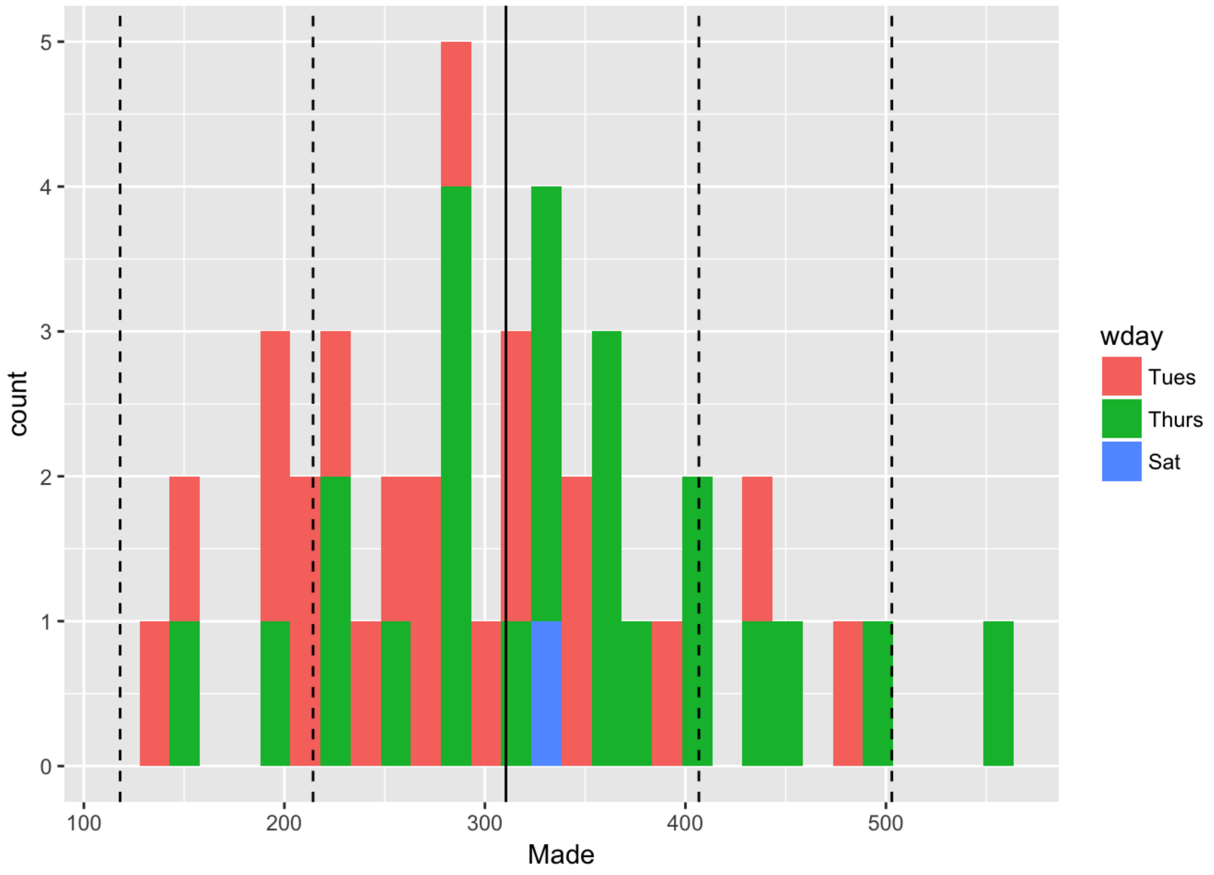
```
garfs %>%
    group_by(Shift) %>%
    summarise(count = n(), percent_n = round((count / nrow(garfs) * 100), digits = 1), mean(Made), median
(Made))
```

Shift <chr>	count <int>	percent_n <dbl>	mean(Made) <dbl>	median(Made) <dbl>
am	29	8.0	309.3103	302.0
am bar	1	0.3	331.0000	331.0
apres	1	0.3	117.0000	117.0
day swing	1	0.3	339.0000	339.0
dub	9	2.5	509.0000	544.0
pm	262	72.0	198.5840	189.0
pm bar	43	11.8	309.0930	308.0
swing	18	4.9	456.7222	439.5

8 rows

```
garfs %>%
    filter(Shift == "pm bar" | Shift == "am bar") %>%
    ggplot(aes(Made)) +
    geom_histogram(aes(fill = wday)) +
    geom_vline(aes(xintercept = median(Made))) +
    geom_vline(aes(xintercept = median(Made) + sd(Made)), linetype = 2) +
    geom_vline(aes(xintercept = median(Made) - sd(Made)), linetype = 2) +
    geom_vline(aes(xintercept = median(Made) + (2 * sd(Made))), linetype = 2) +
    geom_vline(aes(xintercept = median(Made) - (2* sd(Made))), linetype = 2)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

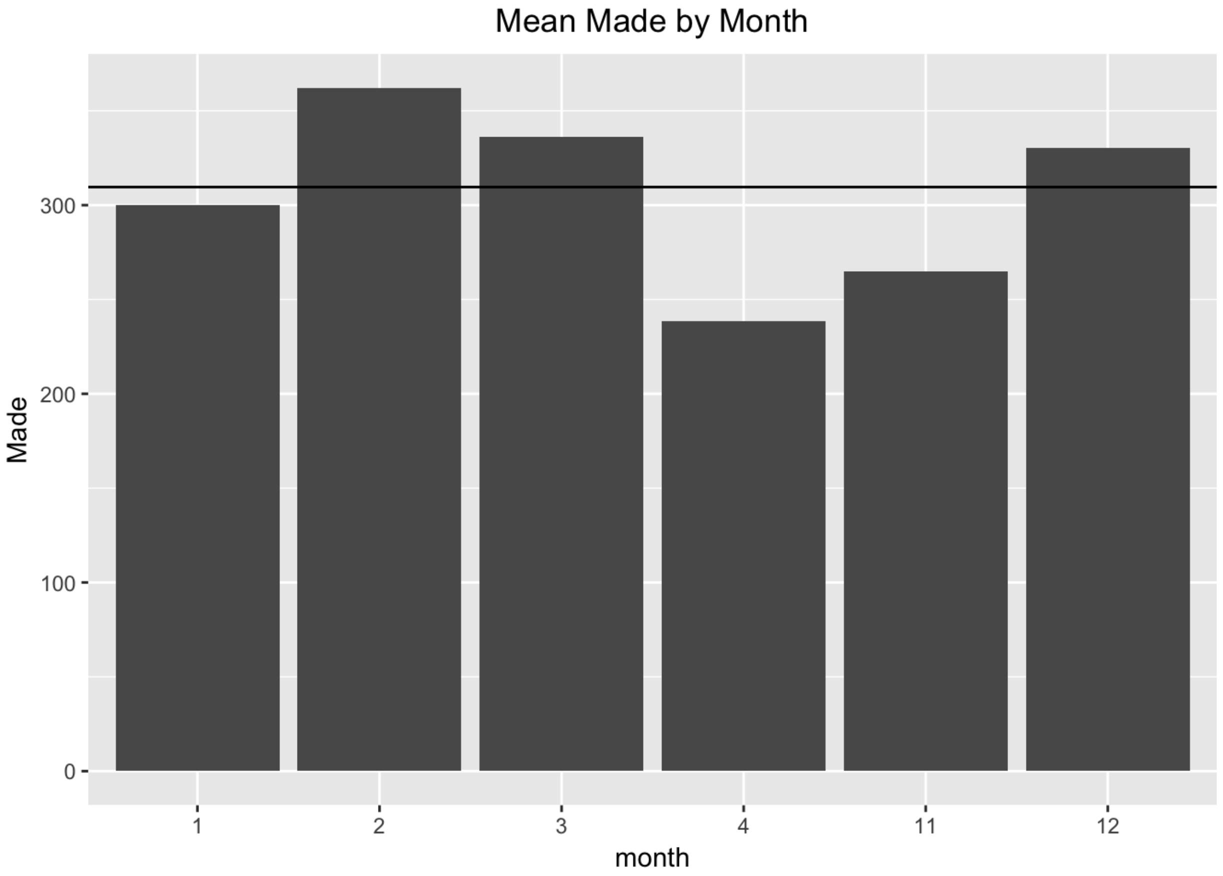


```
garfs %>%
    filter(Shift == "pm bar" | Shift == "am bar") %>%
    group_by(wday) %>%
    summarise(count = n(), mean(Made), median(Made))
```

wday <ord>	count <int>	mean(Made) <dbl>	median(Made) <dbl>
Tues	20	279.400	266.5
Thurs	23	334.913	330.0
Sat	1	331.000	331.0

3 rows

```
garfs %>%
    filter(Shift == "pm bar" | Shift == "am bar") %>%
    ggplot(aes(month, Made)) +
    stat_summary(fun.y = mean, geom = "bar", position = "dodge") +
    geom_hline(aes(yintercept = mean(Made))) +
    ggtitle("Mean Made by Month")
```



```
garfs %>%
    filter(Shift == "pm bar" | Shift == "am bar") %>%
    group_by(year, month) %>%
    summarise(sum(Made), median(Made), mean(Made), n()) %>%
    arrange(desc(year), desc(month))
```

year <fctr>	month <fctr>	sum(Made) <int>	median(Made) <dbl>	mean(Made) <dbl>	n() <int>
2019	4	1432	221.5	238.6667	6
2019	3	2688	323.5	336.0000	8
2019	2	2534	353.0	362.0000	7
2019	1	3001	305.0	300.1000	10
2018	12	2643	324.0	330.3750	8
2018	11	1324	226.0	264.8000	5

6 rows

A look at Winter 18-19

```
garfs %>%
    filter((year == 2018 & month %in% c(11, 12)) | (year == 2019 & month %in% c(1, 2, 3, 4))) %>%
    summarise(median(Made), mean(Made), n(), sum(Made))
```

median(Made) <dbl>	mean(Made) <dbl>	n() <int>	sum(Made) <int>
288.5	295.9298	114	33736

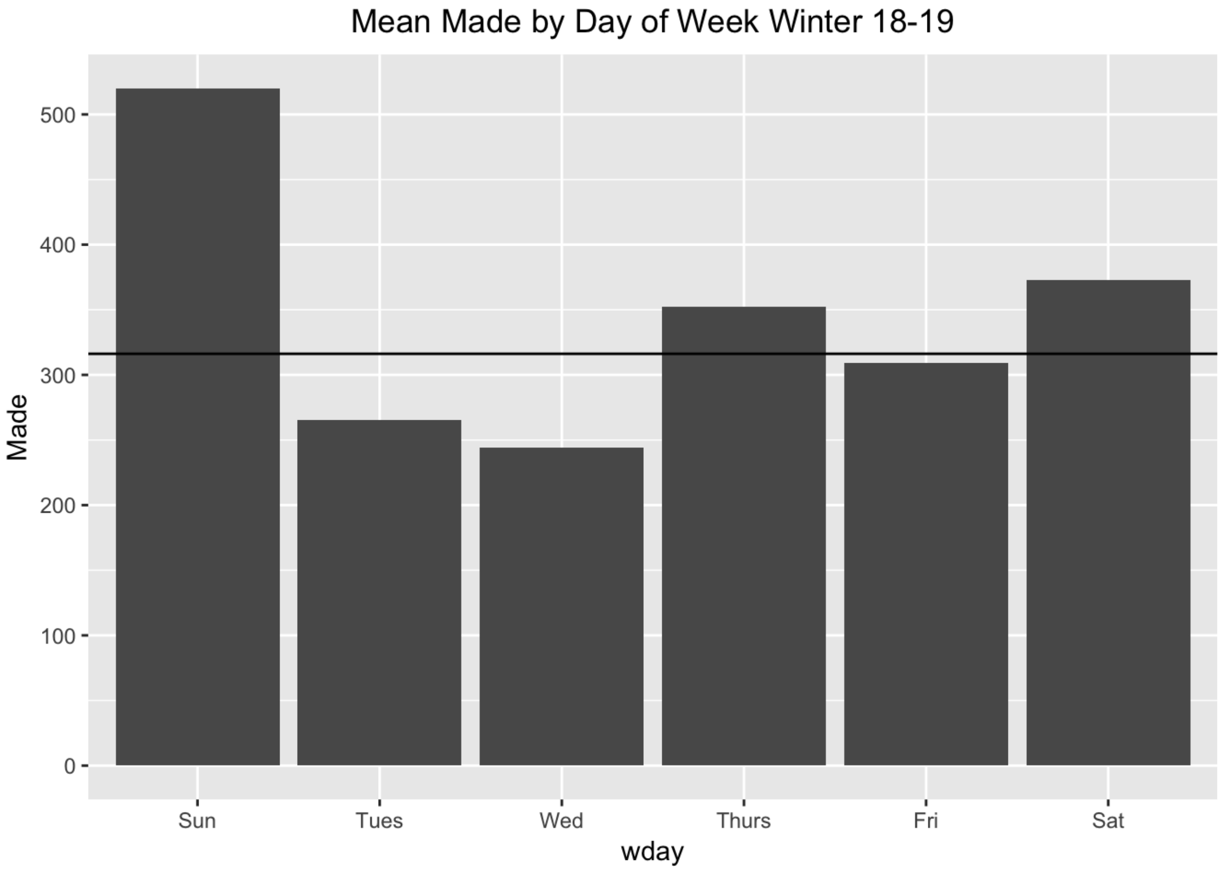
1 row

```
garfs %>%
    filter((year == 2018 & month %in% c(12)) | (year == 2019 & month %in% c(1, 2, 3, 4))) %>%
    summarise(median(Made), mean(Made), n(), sum(Made))
```

median(Made) <dbl>	mean(Made) <dbl>	n() <int>	sum(Made) <int>
305	316.1979	96	30355

1 row

```
garfs %>%
    filter((year == 2018 & month %in% c(12)) | (year == 2019 & month %in% c(1, 2, 3, 4))) %>%
    group_by(wday) %>%
    ggplot(aes(wday, Made)) +
    stat_summary(fun.y = mean, geom = "bar", position = "dodge") +
    geom_hline(aes(yintercept = mean(Made))) +
    ggtitle("Mean Made by Day of Week Winter 18-19")
```

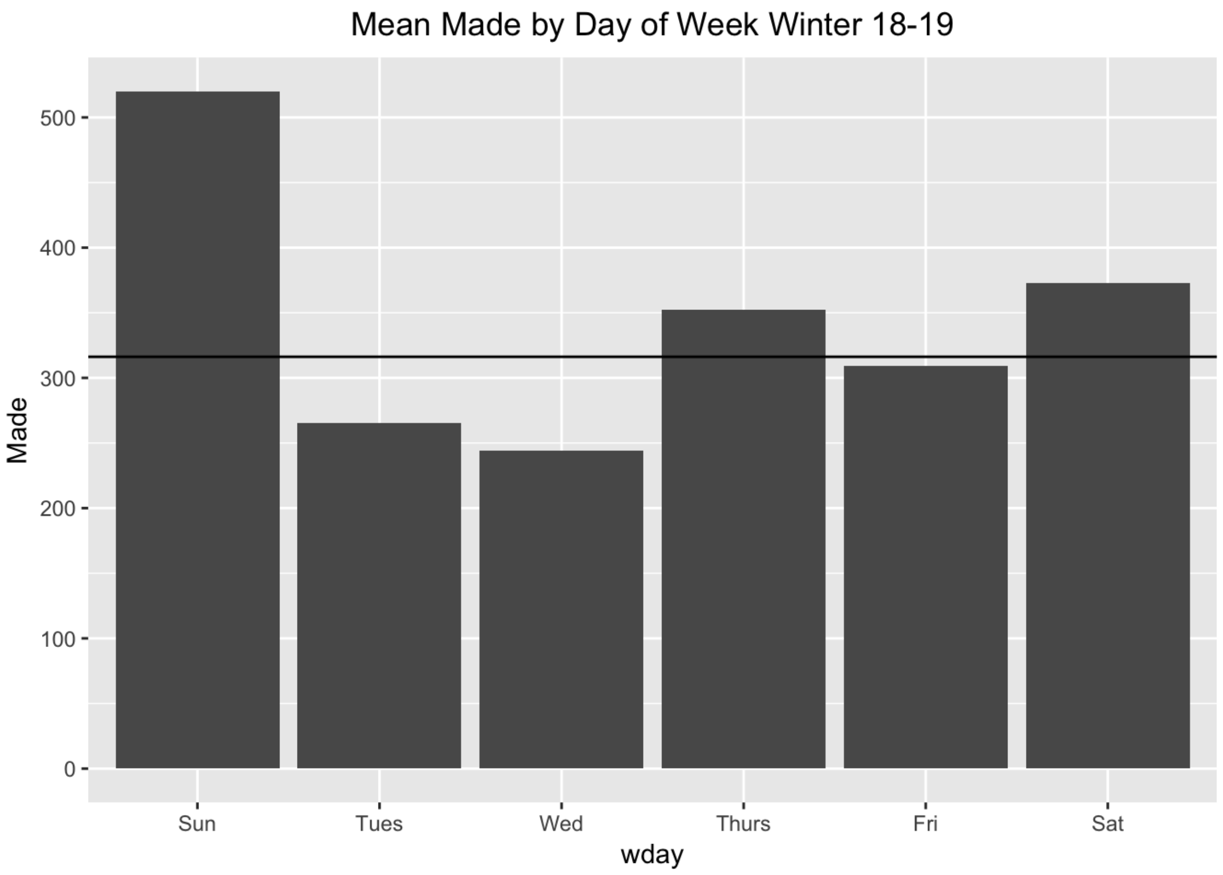


```
garfs %>%
    filter((year == 2018 & month %in% c(12)) | (year == 2019 & month %in% c(1, 2, 3, 4))) %>%
    group_by(wday) %>%
    summarise(median(Made), mean(Made), n(), sum(Made))
```

wday	**median(Made)**	**mean(Made)**	**n()**	**sum(Made)**
<ord>	<dbl>	<dbl>	<int>	<int>
Sun	520	520.0000	2	1040
Tues	266	265.1053	19	5037
Wed	253	244.0625	16	3905
Thurs	344	352.3500	20	7047
Fri	276	309.1053	19	5873
Sat	383	372.6500	20	7453

6 rows

```
garfs %>%
    filter((year == 2018 & month %in% c(12)) | (year == 2019 & month %in% c(1, 2, 3, 4))) %>%
    group_by(wday) %>%
    ggplot(aes(wday, Made)) +
    stat_summary(fun.y = mean, geom = "bar", position = "dodge") +
    geom_hline(aes(yintercept = mean(Made))) +
    ggtitle("Mean Made by Day of Week Winter 18-19")
```



```
garfs %>%
    filter((year == 2018 & month %in% c(12)) | (year == 2019 & month %in% c(1, 2, 3, 4))) %>%
    group_by(wday) %>%
    summarise(median(Made), mean(Made), n(), sum(Made))
```

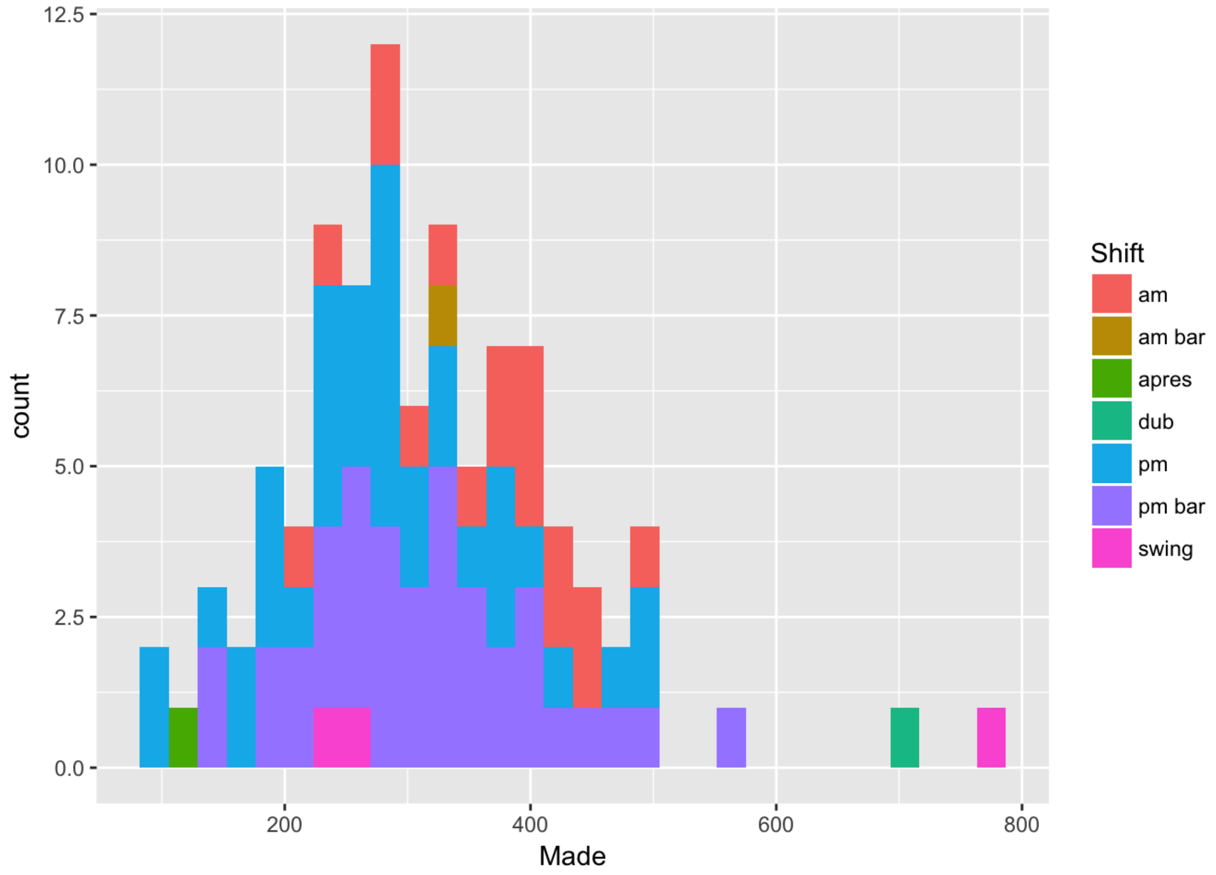
wday	median(Made)	mean(Made)	n()	sum(Made)
<ord>	<dbl>	<dbl>	<int>	<int>
Sun	520	520.0000	2	1040
Tues	266	265.1053	19	5037
Wed	253	244.0625	16	3905
Thurs	344	352.3500	20	7047
Fri	276	309.1053	19	5873
Sat	383	372.6500	20	7453

6 rows

Distribution of Made Winter 18-19

```
garfs %>%
    filter((year == 2018 & month %in% c(12)) | (year == 2019 & month %in% c(1, 2, 3, 4))) %>%
    ggplot(aes(Made, fill = Shift)) +
    geom_histogram()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



‘’’‘’’‘’’‘’’‘’’

A look at Summer 2018

```
garfs %>%
    filter(year == 2018, month %in% c(5, 6, 7, 8, 9)) %>%
    summarise(median(Made), mean(Made), n(), sum(Made))
```

median(Made) <int>	mean(Made) <dbl>	n() <int>	sum(Made) <int>
245	245.2769	65	15943

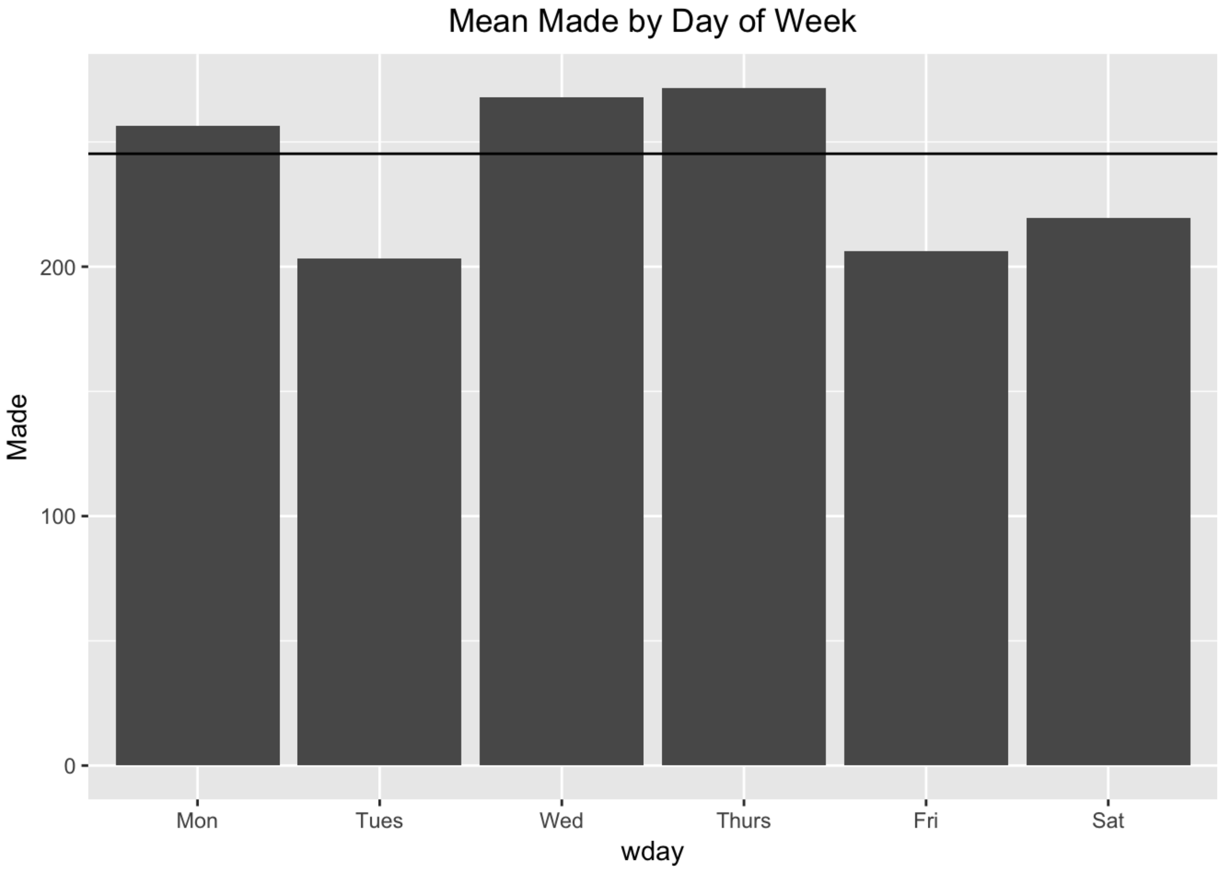
1 row

```
garfs %>%
    filter(year == 2018, month %in% c(5, 6, 7, 8)) %>%
    summarise(median(Made), mean(Made), n(), sum(Made))
```

median(Made) <dbl>	mean(Made) <dbl>	n() <int>	sum(Made) <int>
252	262.7407	54	14188

1 row

```
garfs %>%
    filter(year == 2018, month %in% c(5, 6, 7, 8, 9)) %>%
    group_by(wday) %>%
    ggplot(aes(wday, Made)) +
    stat_summary(fun.y = mean, geom = "bar", position = "dodge") +
    geom_hline(aes(yintercept = mean(Made))) +
    ggtitle("Mean Made by Day of Week")
```

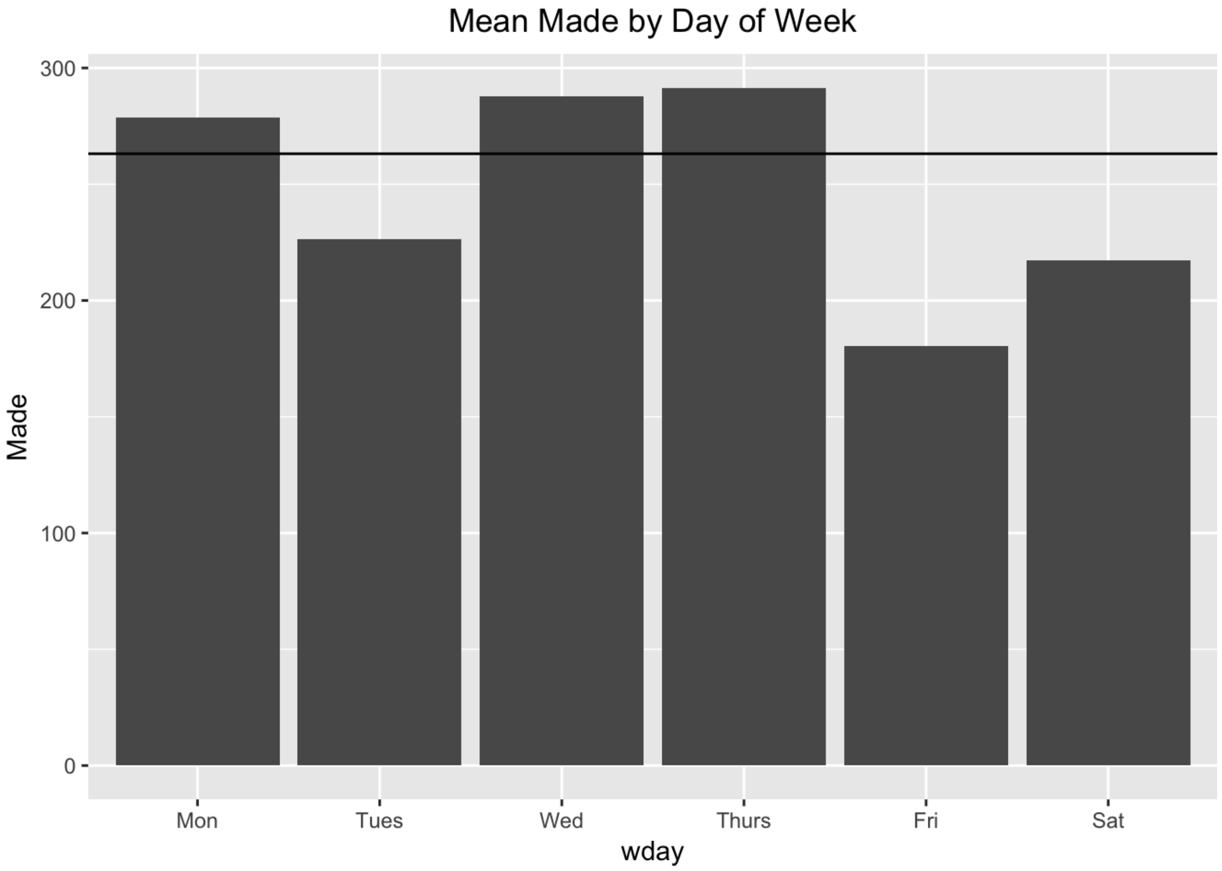


```
garfs %>%
    filter(year == 2018, month %in% c(5, 6, 7, 8, 9)) %>%
    group_by(wday) %>%
    summarise(median(Made), mean(Made), n(), sum(Made))
```

wday <ord>	median(Made) <dbl>	mean(Made) <dbl>	n() <int>	sum(Made) <int>
Mon	263.0	256.3846	13	3333
Tues	187.5	203.2857	14	2846
Wed	243.5	268.0714	14	3753
Thurs	249.0	271.6667	15	4075
Fri	218.0	206.3333	3	619
Sat	224.0	219.5000	6	1317

6 rows

```
garfs %>%
    filter(year == 2018, month %in% c(6, 7, 8)) %>%
    group_by(wday) %>%
    ggplot(aes(wday, Made)) +
    stat_summary(fun.y = mean, geom = "bar", position = "dodge") +
    geom_hline(aes(yintercept = mean(Made))) +
    ggtitle("Mean Made by Day of Week")
```



```
garfs %>%
    filter(year == 2018, month %in% c(6, 7, 8)) %>%
    group_by(wday) %>%
    summarise(median(Made), mean(Made), n(), sum(Made))
```

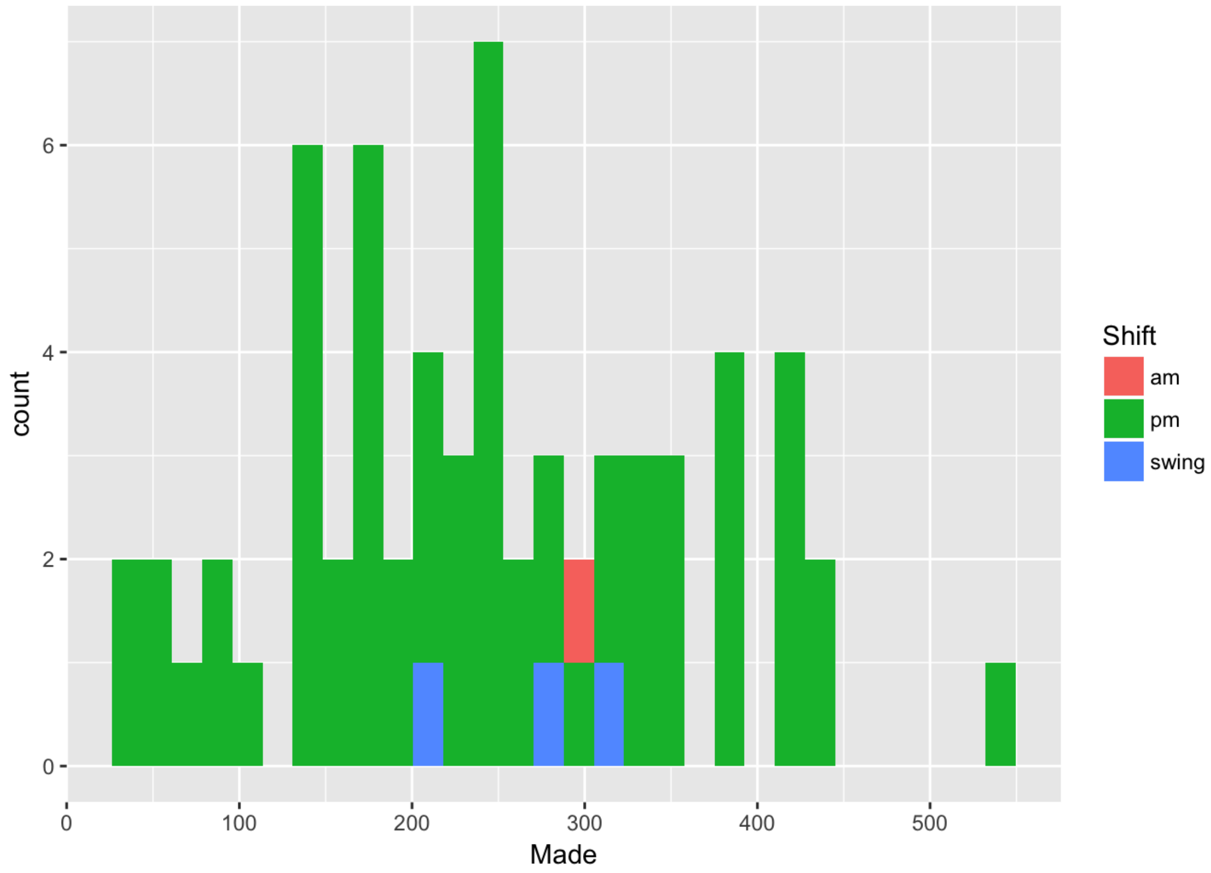
wday <ord>	median(Made) <dbl>	mean(Made) <dbl>	n() <int>	sum(Made) <int>
Mon	286.5	278.5000	10	2785
Tues	210.0	226.4167	12	2717
Wed	261.5	287.7500	12	3453
Thurs	296.0	291.4167	12	3497
Fri	180.5	180.5000	2	361
Sat	222.0	217.2500	4	869

6 rows

Distribution of Made during Summer 2018

```
garfs %>%
    filter(year == 2018, month %in% c(5, 6, 7, 8, 9)) %>%
    ggplot(aes(Made, fill = Shift)) +
    geom_histogram()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



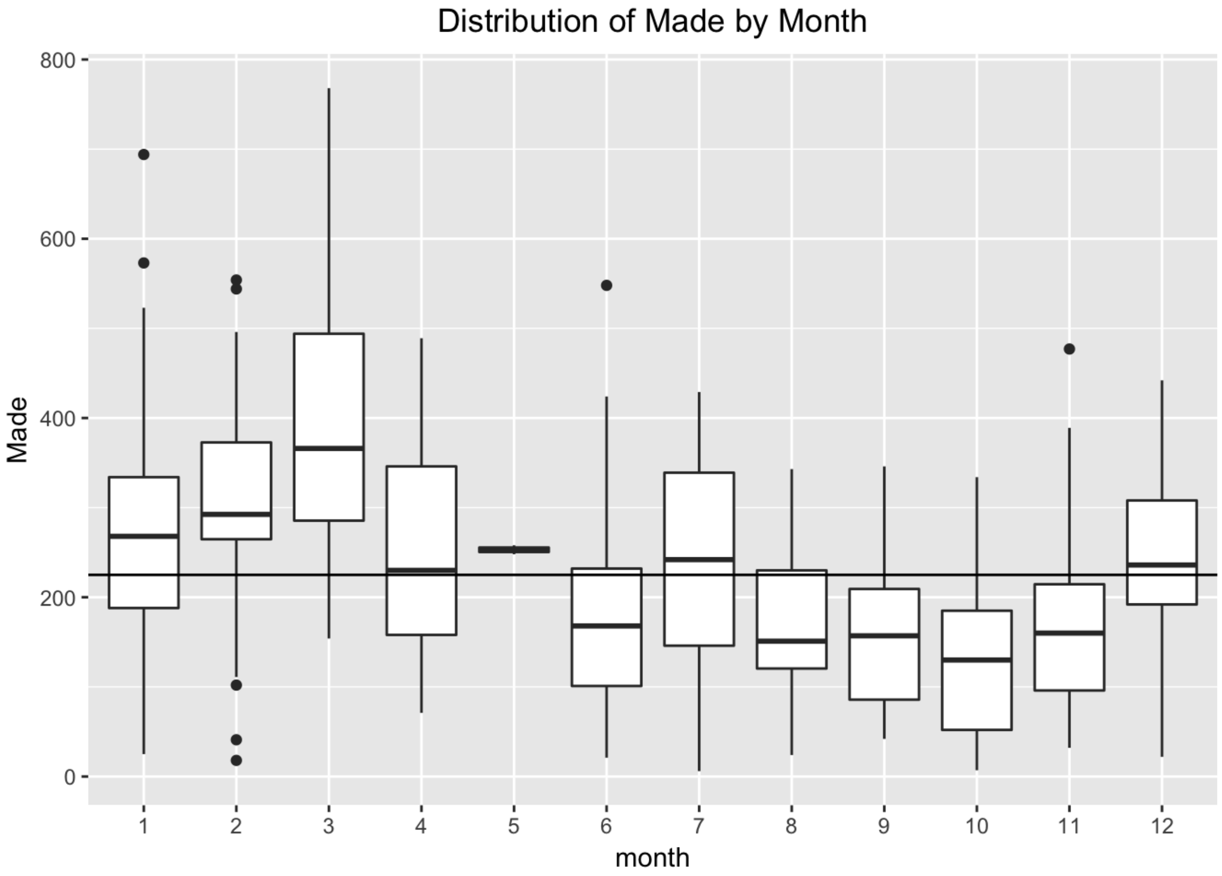
Total sales since I began tracking

```
garfs %>%
    summarise(sum(LittleNumber, na.rm = TRUE))
```

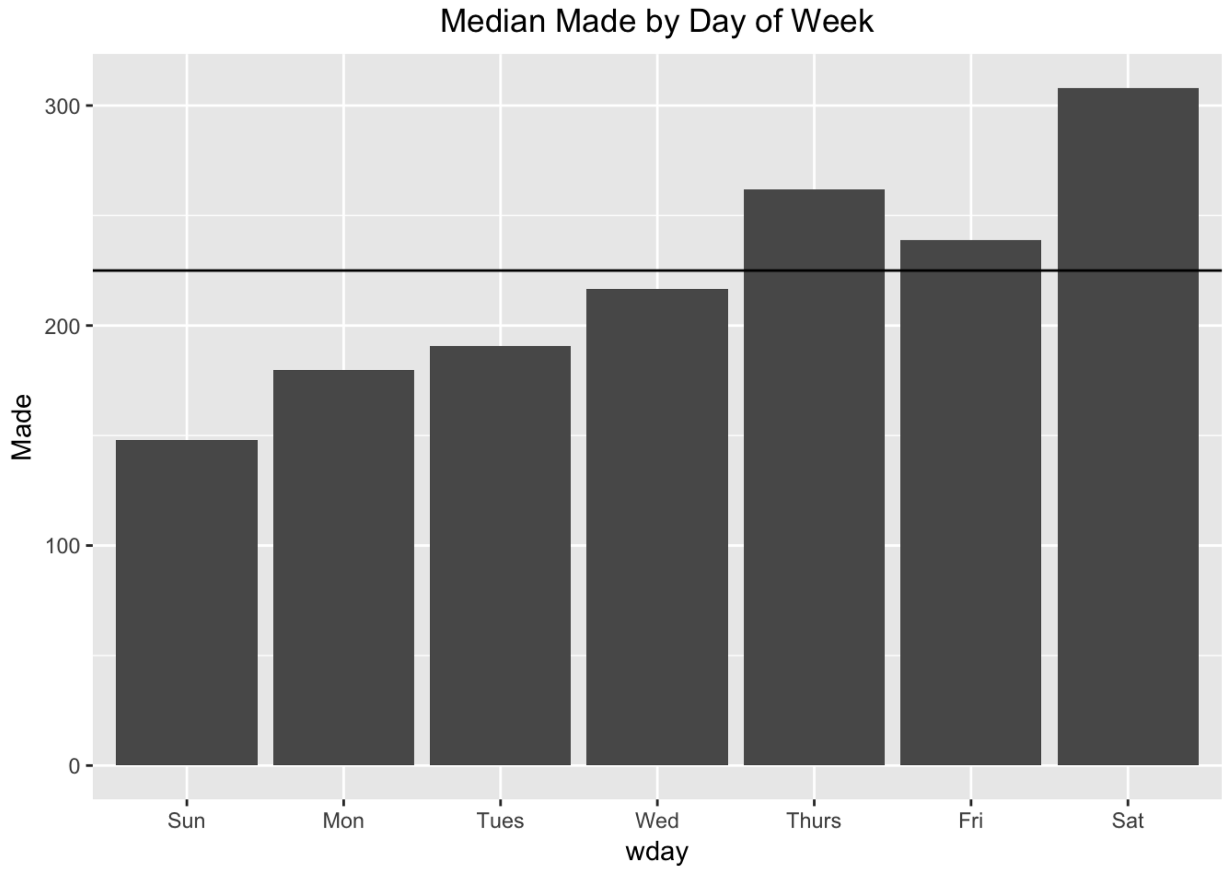
sum(LittleNumber, na.rm = TRUE) <dbl>
431244.8

1 row

```
garfs %>%
    ggplot(aes(month, Made)) +
    geom_boxplot() +
    geom_hline(aes(yintercept = median(Made))) +
    ggtitle("Distribution of Made by Month")
```



```
garfs %>%
    ggplot(aes(wday, Made)) +
    stat_summary(fun.y = median, geom = "bar") +
    geom_hline(aes(yintercept = median(Made))) +
    ggtitle("Median Made by Day of Week")
```

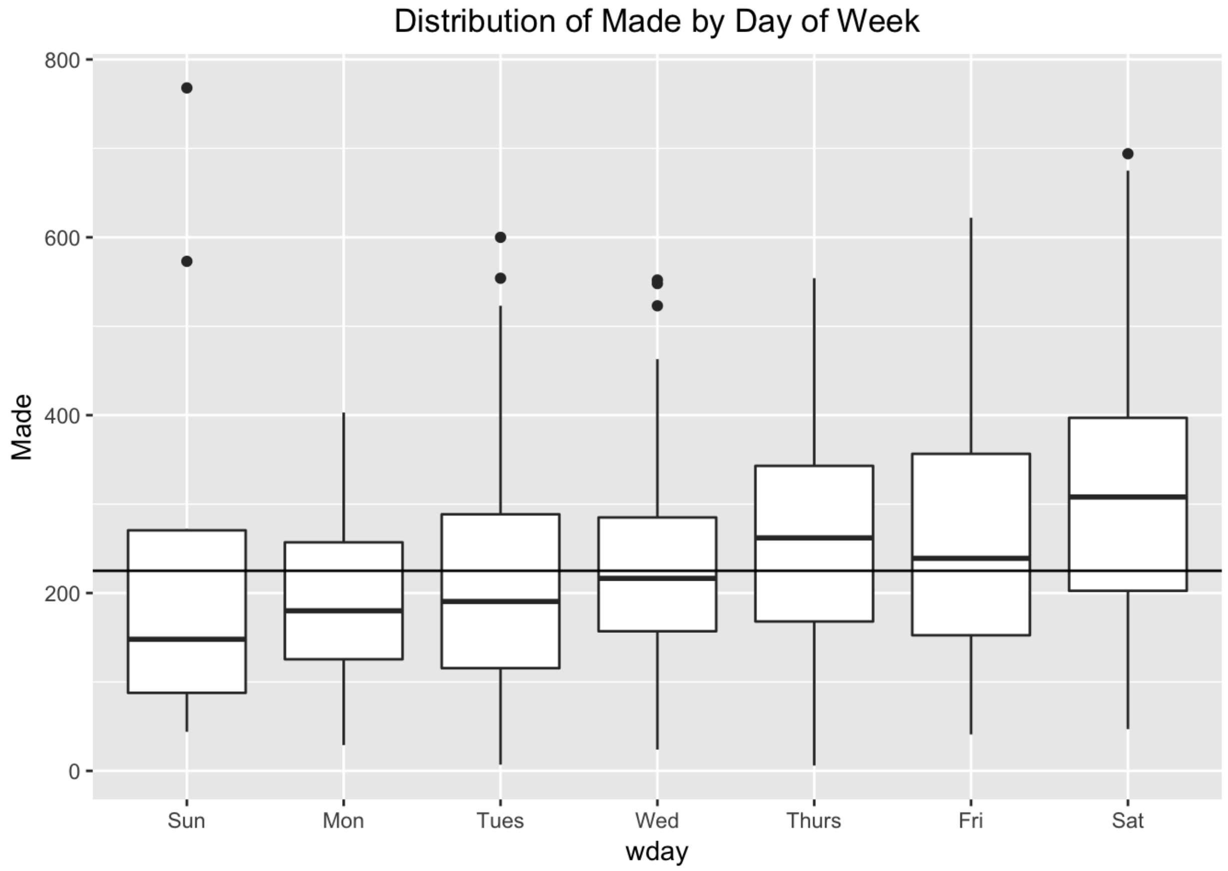


```
garfs %>%
    group_by(wday) %>%
    summarise(median(Made), mean(Made), n())
```

wday	median(Made)	mean(Made)	n()
<ord>	<dbl>	<dbl>	<int>
Sun	148.0	232.7500	12
Mon	180.0	195.6154	39
Tues	190.5	207.1667	78
Wed	216.5	229.5278	72
Thurs	262.0	258.3279	61
Fri	239.0	262.6471	51
Sat	308.0	306.2549	51

7 rows

```
garfs %>%
    ggplot(aes(wday, Made)) +
    geom_boxplot() +
    geom_hline(aes(yintercept = median(Made))) +
    ggtitle("Distribution of Made by Day of Week")
```



When identifying outliers, I will arbitrarily call anything lower than **50** or higher than **300** an outlier.

```
garfs %>%
    filter(Made < 50)
```

Date	BigNum…	LittleNumber	NegD…	Din…	Shift	CutTi…	M…	Comments
<date>	<dbl>	<dbl>	<dbl>	<int>	<chr>	<int>	<int>	<chr>
2017-06-27	174.57	NA	NA	1	pm	6	21	cut early
2017-07-10	379.48	369.50	NA	0	pm	NA	48	NA
2017-07-13	55.30	71.00	-11.00	0	pm	5	6	cut early
2017-08-23	158.28	124.00	NA	0	pm	6	24	cut early
2017-10-01	333.95	308.03	NA	0	pm	6	44	cut early
2017-11-28	345.29	357.00	NA	1	pm	NA	49	NA
2017-12-02	336.36	348.25	-56.49	1	pm	7	47	cut early
2017-12-05	75.89	70.00	NA	0	pm	6	22	cut earluy
2018-01-16	149.88	144.25	NA	0	pm	6	25	cut super early
2018-02-06	124.68	115.00	-24.80	0	pm	6	18	cut super early

1-10 of 20 rows | 1-9 of 15 columns Previous **1** 2 Next

```
garfs %>%
    filter(Made > 300)
```

Date	BigNumber	LittleNumber	NegDrop	Dinner	Shift	CutTime	Made
<date>	<dbl>	<dbl>	<dbl>	<int>	<chr>	<int>	<int>
2017-07-01	1756.57	1624.00	NA	0	pm	NA	347
2017-09-13	1746.86	1570.25	-158.83	0	pm	NA	316
2017-11-21	1539.68	1420.25	-307.78	0	pm	NA	346
2017-12-29	1867.13	1761.00	NA	0	pm	NA	347
2018-01-02	3775.63	3506.75	-278.64	1	dub	NA	523
2018-01-05	2968.32	2755.00	NA	0	dub	NA	413
2018-01-06	1994.19	1814.50	NA	0	pm	NA	326
2018-01-12	2591.36	2417.50	NA	0	dub	NA	334
2018-01-13	2026.43	1880.25	-97.62	0	pm	NA	480
2018-01-21	3735.21	3561.25	-34.66	0	dub	NA	573

1-10 of 109 rows | 1-8 of 15 columns Previous **1** 2 3 4 5 6 ... 11 Next

Now a look at how much I made as a **percentage of sales**:

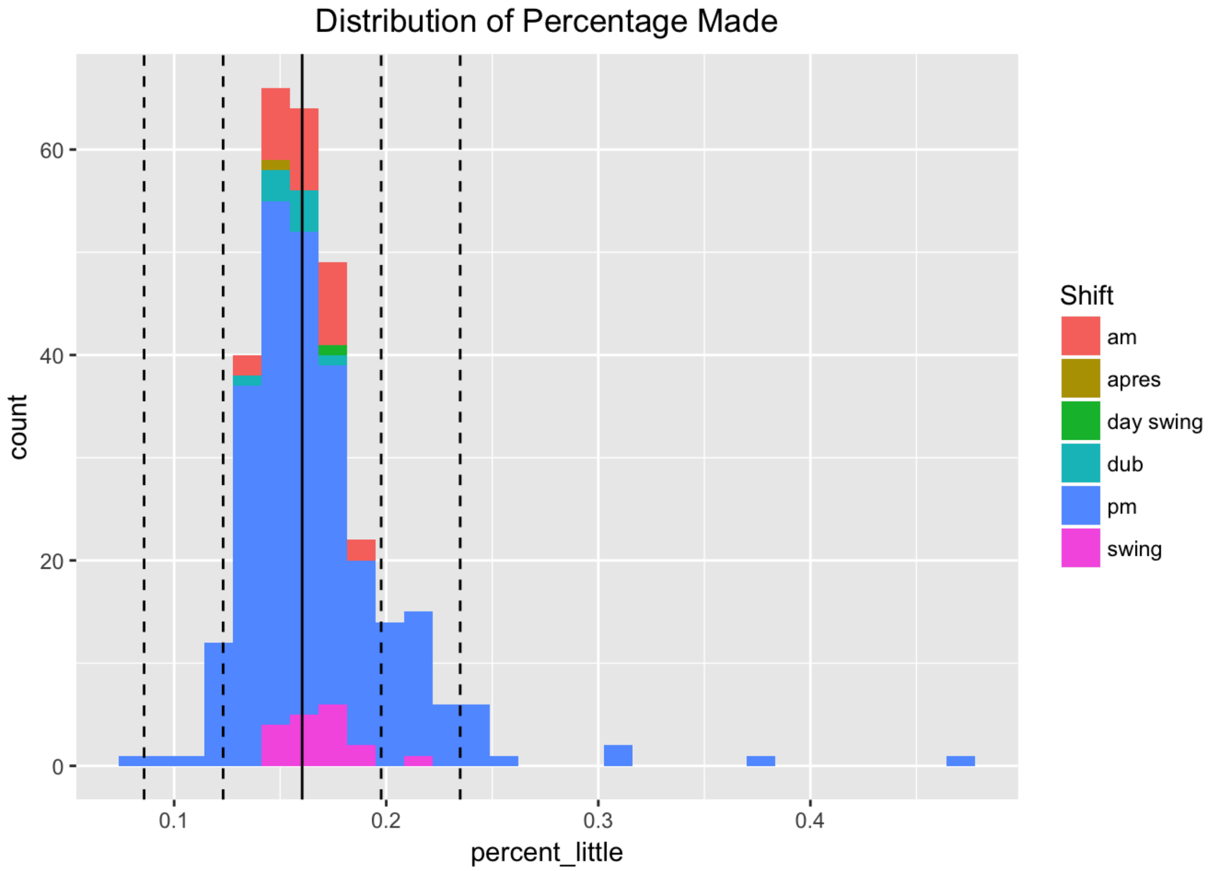
```
summary(garfs$percent_little)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
## 0.08451 0.14510 0.16040 0.16750 0.17960 0.47480      62
```

```
garfs %>%
    ggplot(aes(percent_little)) +
    geom_histogram(aes(fill = Shift)) +
    geom_vline(aes(xintercept = median(percent_little, na.rm = TRUE))) +
    geom_vline(aes(xintercept = median(percent_little, na.rm = TRUE) + sd(percent_little, na.rm = TRUE)),
linetype = 2) +
    geom_vline(aes(xintercept = median(percent_little, na.rm = TRUE) - sd(percent_little, na.rm = TRUE)),
linetype = 2) +
    geom_vline(aes(xintercept = median(percent_little, na.rm = TRUE) + (2 * sd(percent_little, na.rm = TR
UE))), linetype = 2) +
    geom_vline(aes(xintercept = median(percent_little, na.rm = TRUE) - (2 * sd(percent_little, na.rm = TR
UE))), linetype = 2) +
    ggtitle("Distribution of Percentage Made")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 62 rows containing non-finite values (stat_bin).
```



```
garfs %>%
    summarise(median(percent_little, na.rm = TRUE), mean(percent_little, na.rm = TRUE))
```

median(percent_little, na.rm = TRUE)	mean(percent_little, na.rm = TRUE)
<dbl>	<dbl>
0.1603526	0.1674944

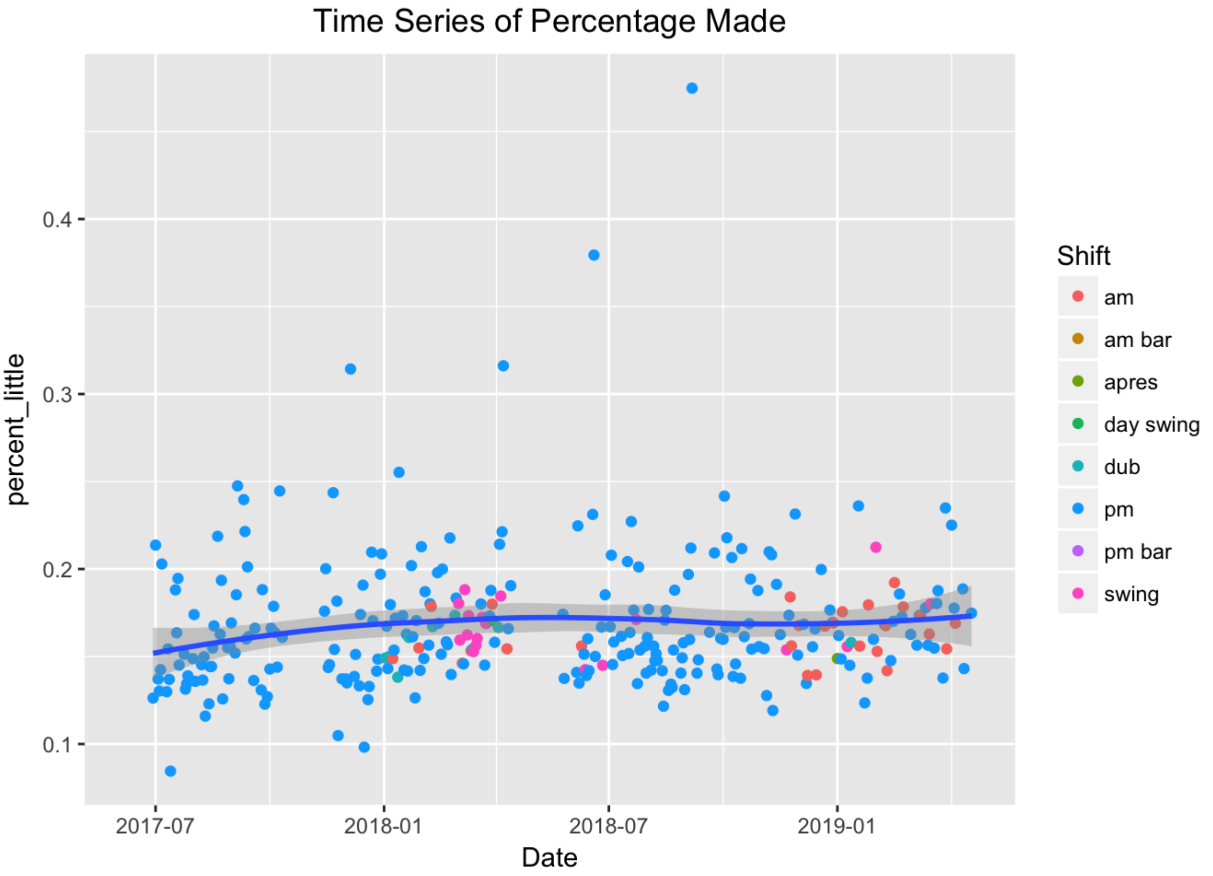
1 row

```
garfs %>%
    ggplot(aes(Date, percent_little)) +
    geom_point(aes(color = Shift)) +
    geom_smooth() +
    ggtitle("Time Series of Percentage Made")
```

```
## `geom_smooth()` using method = 'loess'
```

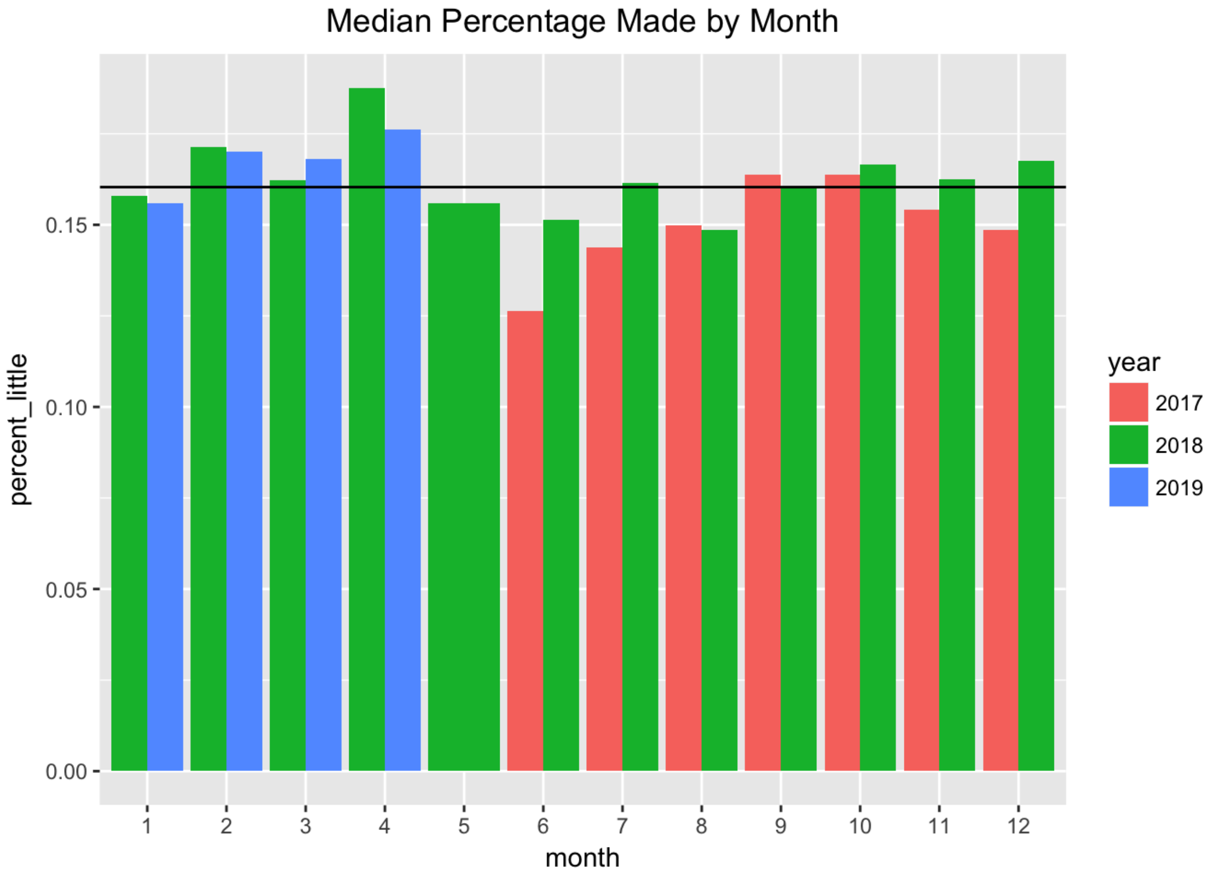
```
## Warning: Removed 62 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 62 rows containing missing values (geom_point).
```



```
garfs %>%
    ggplot(aes(month, percent_little, fill = year)) +
    stat_summary(fun.y = median, geom = "bar", position = "dodge") +
    geom_hline(aes(yintercept = median(percent_little, na.rm = TRUE))) +
    ggtitle("Median Percentage Made by Month")
```

```
## Warning: Removed 62 rows containing non-finite values (stat_summary).
```



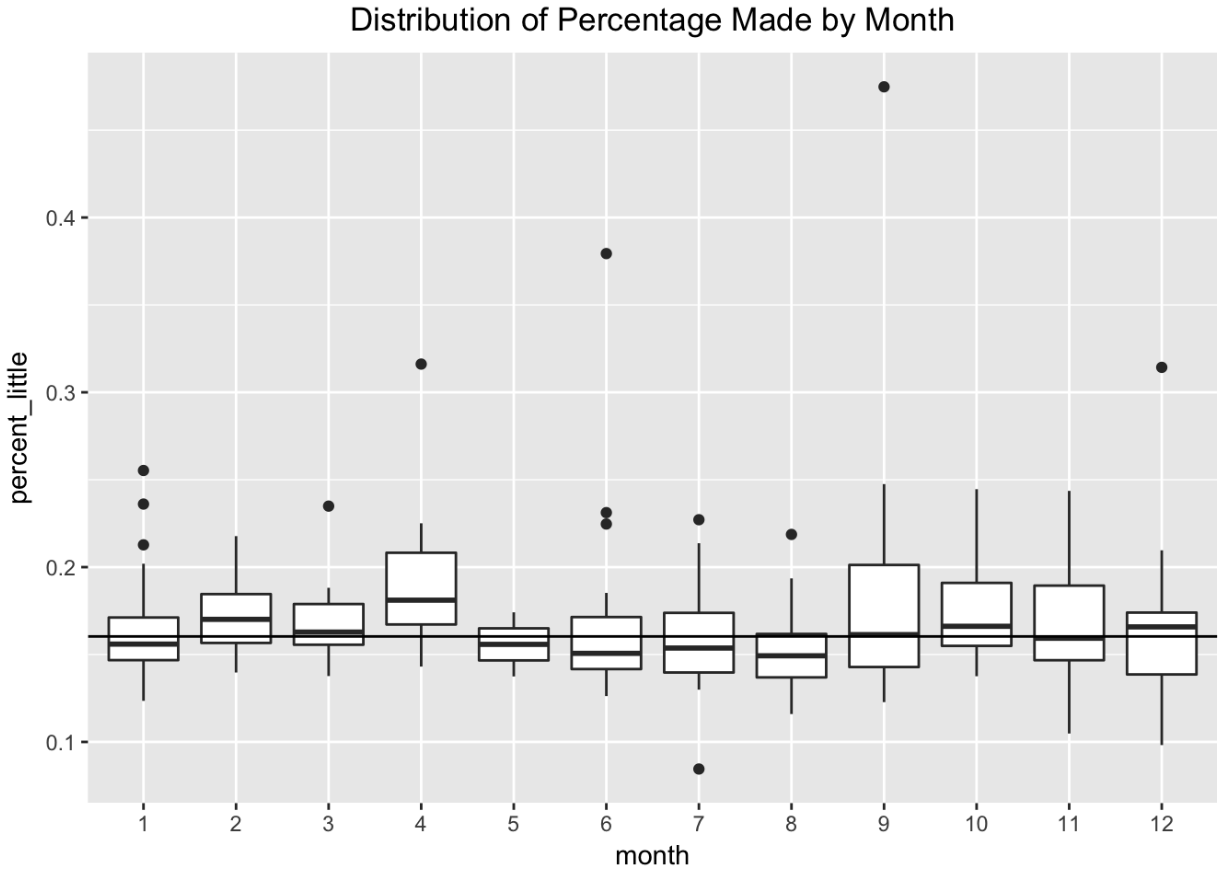
```
garfs %>%
    group_by(year, month) %>%
    summarise(Median = median(percent_little, na.rm = TRUE), Mean = mean(percent_little, na.rm = TRUE))
```

year <fctr>	month <fctr>	Median <dbl>	Mean <dbl>
2017	6	0.1262425	0.1262425
2017	7	0.1437880	0.1515753
2017	8	0.1498751	0.1531407
2017	9	0.1637862	0.1742878
2017	10	0.1637902	0.1715578
2017	11	0.1540578	0.1651777
2017	12	0.1485302	0.1651019
2018	1	0.1578871	0.1639887
2018	2	0.1712389	0.1728709
2018	3	0.1623540	0.1651489

1-10 of 23 rows Previous **1** 2 3 Next

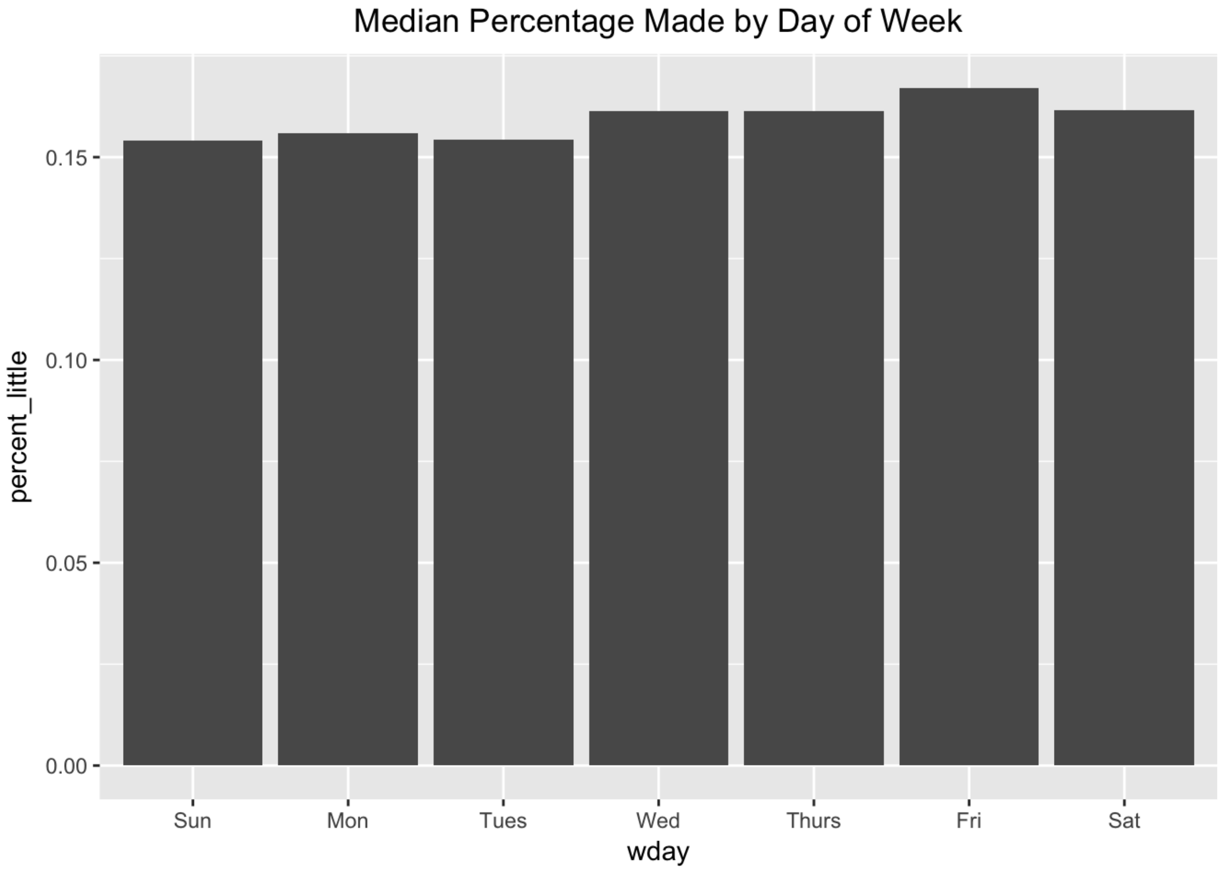
```
garfs %>%
    ggplot(aes(month, percent_little)) +
    geom_boxplot() +
    geom_hline(aes(yintercept = median(percent_little, na.rm = TRUE))) +
    ggtitle("Distribution of Percentage Made by Month")
```

```
## Warning: Removed 62 rows containing non-finite values (stat_boxplot).
```



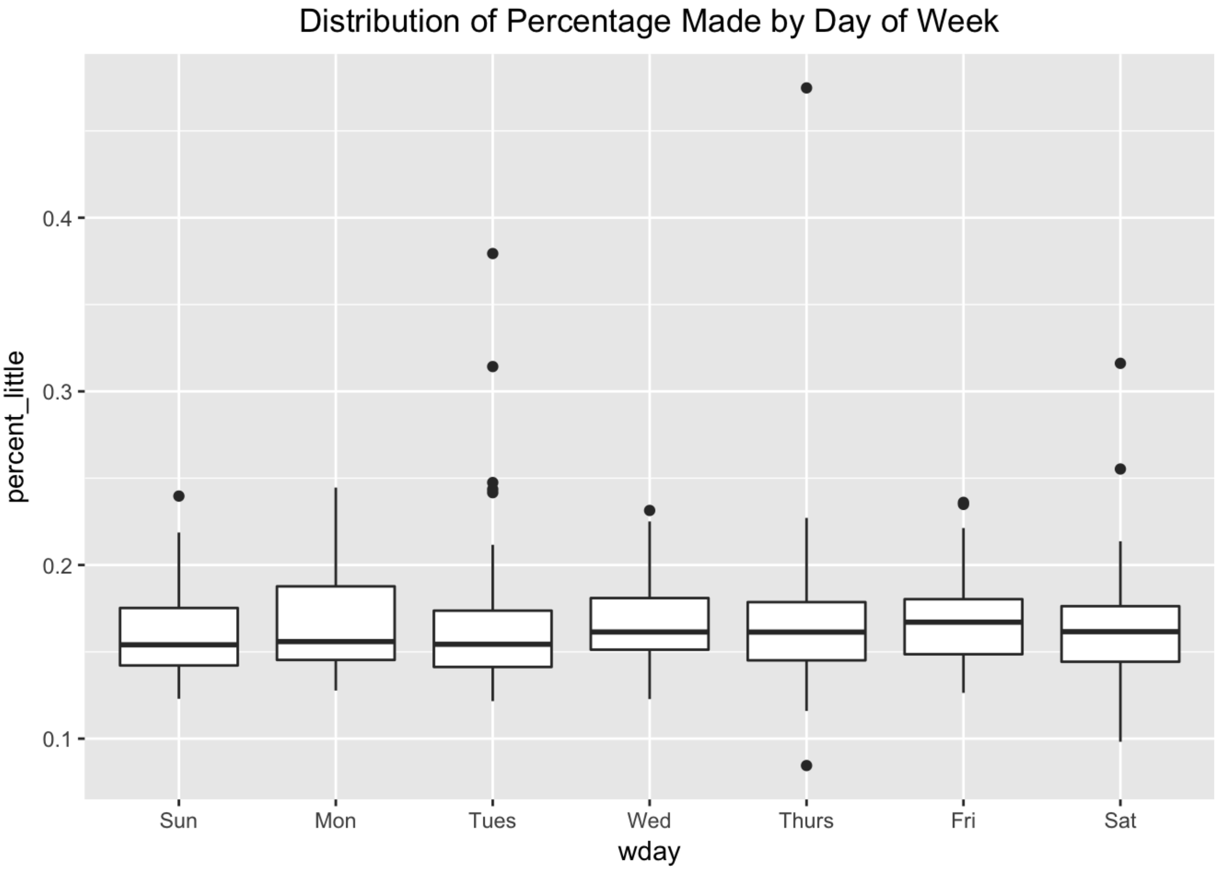
```
garfs %>%
    ggplot(aes(wday, percent_little)) +
    stat_summary(fun.y = median, geom = "bar") +
#    geom_hline(aes(yintercept = median(percent_little, na.rm = TRUE))) +
    ggtitle("Median Percentage Made by Day of Week")
```

```
## Warning: Removed 62 rows containing non-finite values (stat_summary).
```



```
garfs %>%
    ggplot(aes(wday, percent_little)) +
    geom_boxplot() +
#    geom_hline(aes(yintercept = median(percent_little, na.rm = TRUE))) +
    ggtitle("Distribution of Percentage Made by Day of Week")
```

```
## Warning: Removed 62 rows containing non-finite values (stat_boxplot).
```



When identifying outliers, I will arbitrarily call anything lower than **.13** or higher than **.23** an outlier.

```
garfs %>%
    filter(percent_little < .13)
```

Date <date>	BigNum… <dbl>	LittleNumber <dbl>	NegD… <dbl>	Dinner <int>	Shift <chr>	CutTi… <int>	M… <int>	Comments <chr>
2017-06-29	1082.58	1006.00	NA	1	pm	NA	127	NA
2017-07-10	379.48	369.50	NA	0	pm	NA	48	NA
2017-07-13	55.30	71.00	-11.00	0	pm	5	6	cut early
2017-08-10	1160.00	1078.00	-52.06	1	pm	NA	125	NA
2017-08-13	760.64	707.50	-30.81	1	pm	NA	87	NA
2017-08-24	1620.46	1502.75	NA	1	pm	NA	189	NA
2017-09-27	634.25	619.00	-93.67	0	pm	NA	76	NA
2017-09-29	906.83	842.04	-94.22	1	pm	NA	107	NA
2017-11-25	508.45	477.00	-72.52	1	pm	NA	50	less 23 iou (I think)
2017-12-16	601.79	600.75	-12.47	0	pm	NA	59	NA

1-10 of 16 rows | 1-9 of 15 columns Previous **1** 2 Next

```
garfs %>%
    filter(percent_little > .23)
```

Date	BigNumber	LittleNumber	NegDrop	Dinner	Shift	CutTime	Made	▸
<date>	<dbl>	<dbl>	<dbl>	<int>	<chr>	<int>	<int>	
2017-09-05	885.07	824.25	NA	0	pm	NA	204	
2017-09-10	687.86	642.50	NA	1	pm	NA	154	
2017-10-09	427.24	331.17	-79.00	0	pm	NA	81	
2017-11-21	1539.68	1420.25	-307.78	0	pm	NA	346	
2017-12-05	75.89	70.00	NA	0	pm	6	22	
2018-01-13	2026.43	1880.25	-97.62	0	pm	NA	480	
2018-04-07	1188.16	727.50	NA	1	pm	NA	230	
2018-06-18	1745.15	1630.75	-51.00	0	pm	NA	377	
2018-06-19	1179.50	1115.00	-105.56	0	pm	NA	423	
2018-09-06	388.13	366.50	-14.08	1	pm	NA	174	

1-10 of 14 rows | 1-8 of 15 columns Previous **1** 2 Next

Now to remove some of the noise of percentage made by looking at **sales**:

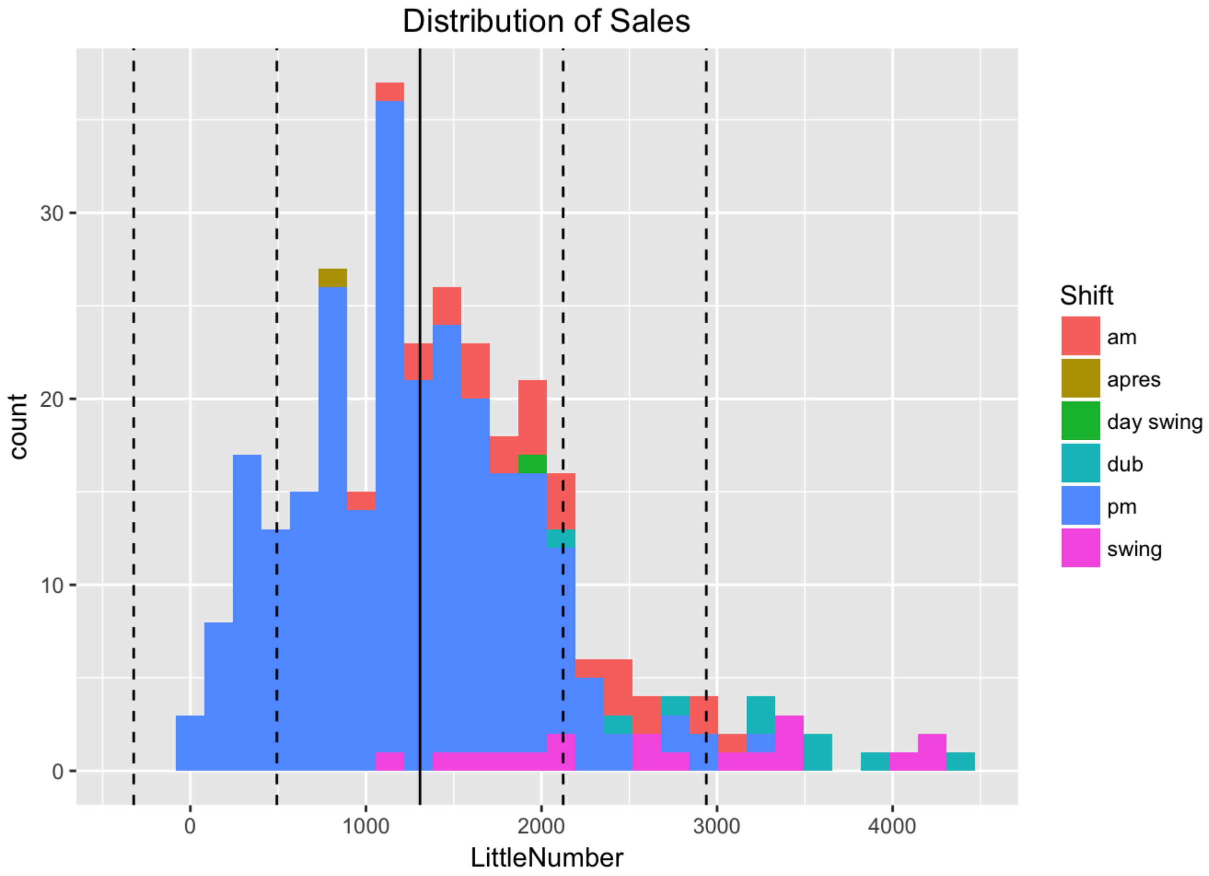
```
summary(garfs$LittleNumber)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
##    50.5   842.0  1308.0  1428.0  1873.0  4392.0      62
```

```
garfs %>%
    ggplot(aes(LittleNumber)) +
    geom_histogram(aes(fill = Shift)) +
    geom_vline(aes(xintercept = median(LittleNumber, na.rm = TRUE))) +
    geom_vline(aes(xintercept = median(LittleNumber, na.rm = TRUE) + sd(LittleNumber, na.rm = TRUE)), lin
etype = 2) +
    geom_vline(aes(xintercept = median(LittleNumber, na.rm = TRUE) - sd(LittleNumber, na.rm = TRUE)), lin
etype = 2) +
    geom_vline(aes(xintercept = median(LittleNumber, na.rm = TRUE) + (2 * sd(LittleNumber, na.rm = TRU
E))), linetype = 2) +
    geom_vline(aes(xintercept = median(LittleNumber, na.rm = TRUE) - (2 * sd(LittleNumber, na.rm = TRU
E))), linetype = 2) +
    ggtitle("Distribution of Sales")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 62 rows containing non-finite values (stat_bin).
```



```
garfs %>%
    summarise(median(LittleNumber, na.rm = TRUE), mean(LittleNumber, na.rm = TRUE))
```

median(LittleNumber, na.rm = TRUE) <dbl>	mean(LittleNumber, na.rm = TRUE) <dbl>
1307.965	1427.963

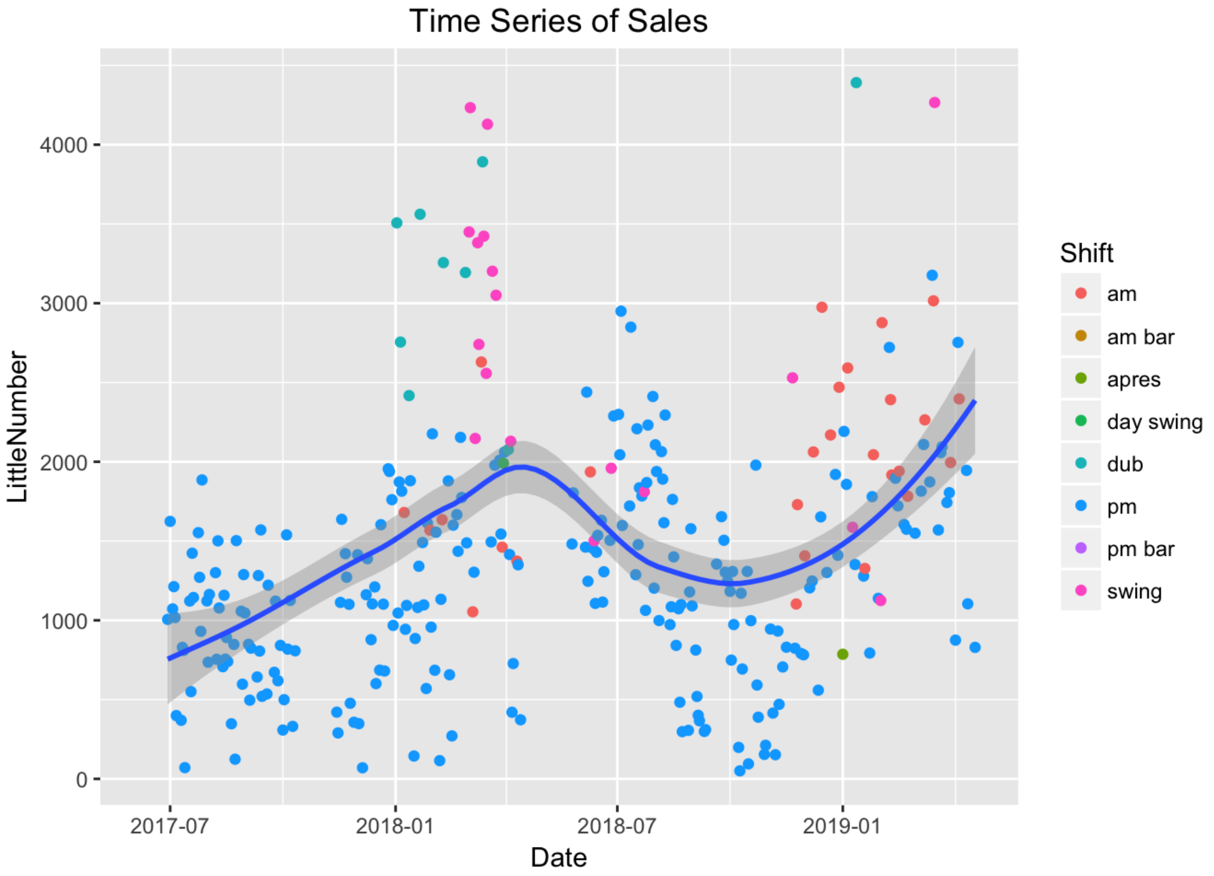
1 row

```
garfs %>%
    ggplot(aes(Date, LittleNumber)) +
    geom_point(aes(color = Shift)) +
    geom_smooth() +
    ggtitle("Time Series of Sales")
```

```
## `geom_smooth()` using method = 'loess'
```

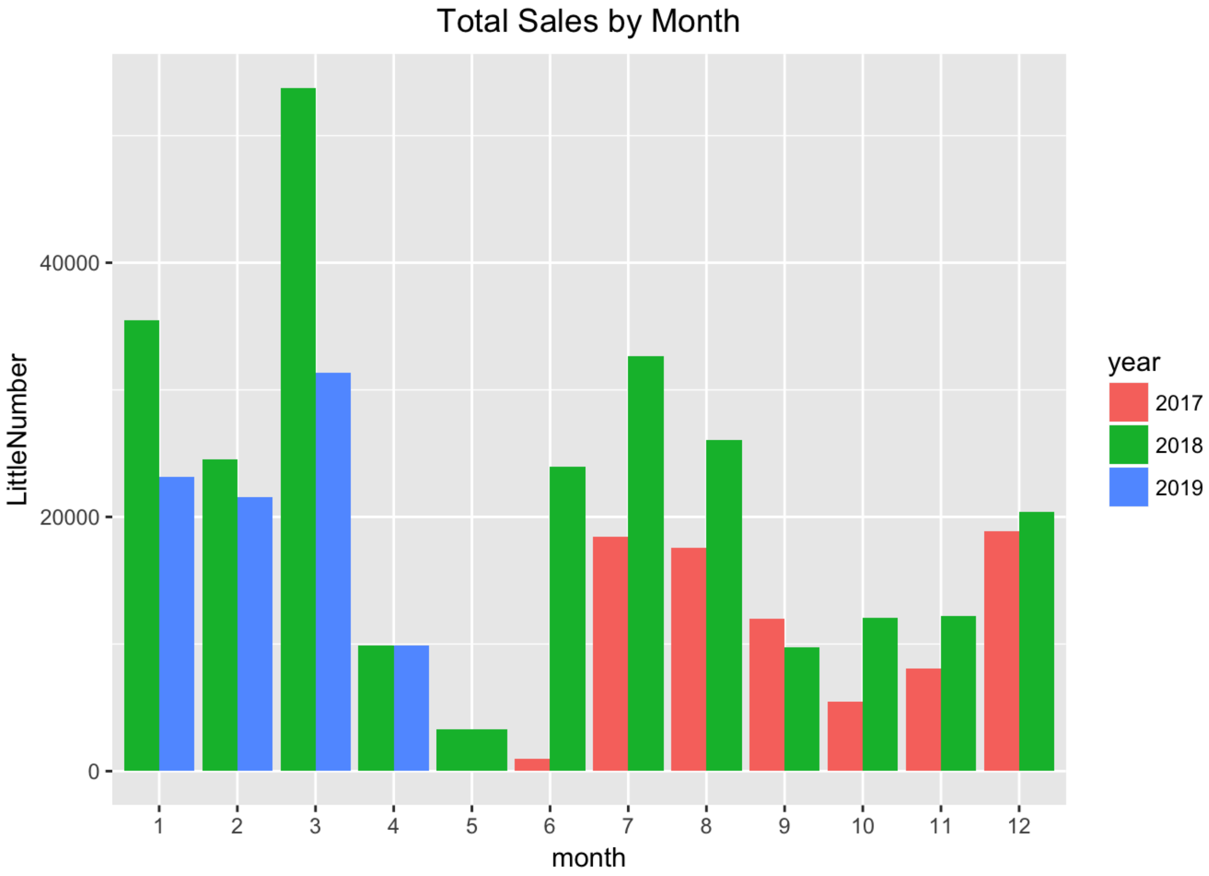
```
## Warning: Removed 62 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 62 rows containing missing values (geom_point).
```



```
garfs %>%
    ggplot(aes(month, LittleNumber, fill = year)) +
    stat_summary(fun.y = sum, geom = "bar", position = "dodge") +
#     geom_hline(aes(yintercept = median(LittleNumber, na.rm = TRUE))) +
    ggtitle("Total Sales by Month")
```

```
## Warning: Removed 62 rows containing non-finite values (stat_summary).
```



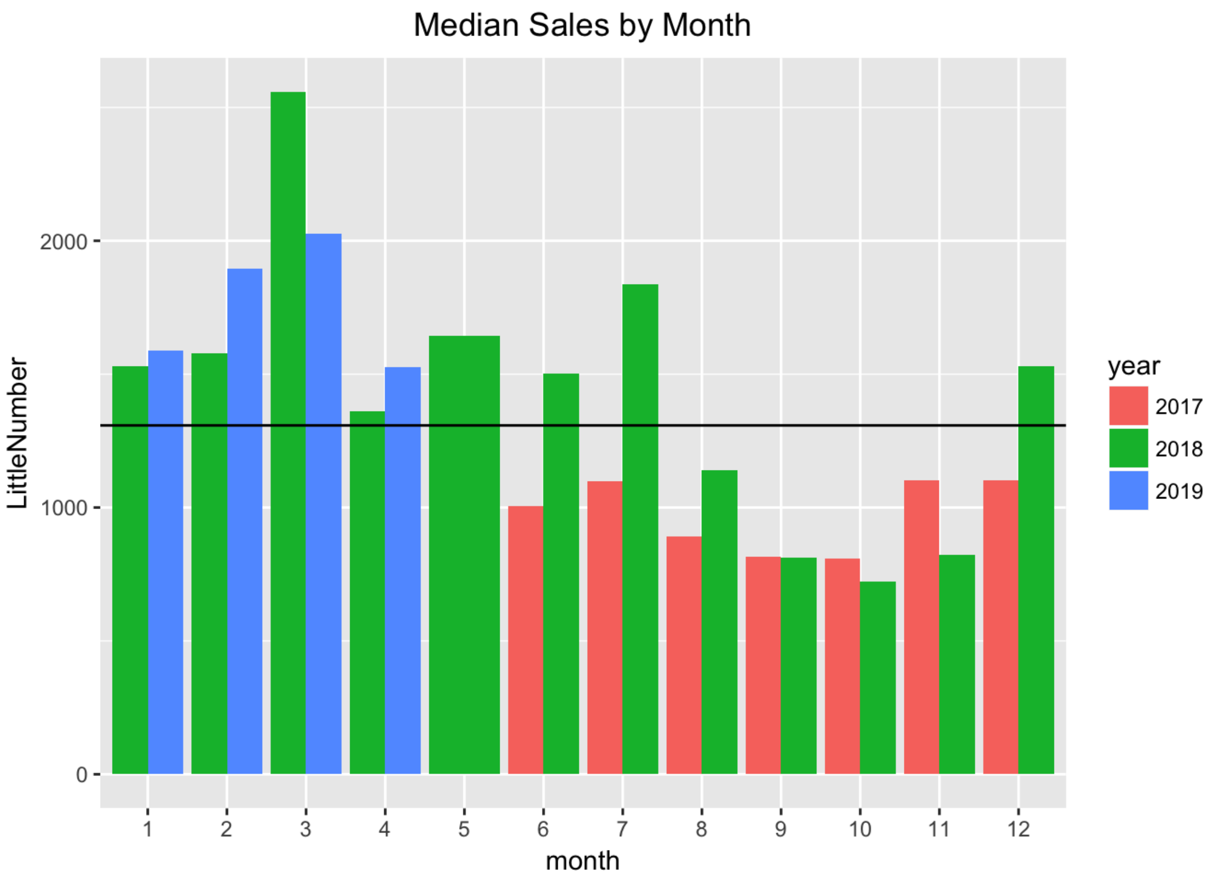
```
garfs %>%
    group_by(year, month) %>%
    summarise(sum(LittleNumber, na.rm = TRUE), n())
```

year <fctr>	month <fctr>	sum(LittleNumber, na.rm = TRUE) <dbl>	n() <int>
2017	6	1006.00	16
2017	7	18407.50	18
2017	8	17597.00	20
2017	9	12002.79	15
2017	10	5429.36	7
2017	11	8088.00	9
2017	12	18870.75	17
2018	1	35496.00	22
2018	2	24500.00	16
2018	3	53732.42	21

1-10 of 23 rows Previous **1** 2 3 Next

```
garfs %>%
    ggplot(aes(month, LittleNumber, fill = year)) +
    stat_summary(fun.y = median, geom = "bar", position = "dodge") +
    geom_hline(aes(yintercept = median(LittleNumber, na.rm = TRUE))) +
    ggtitle("Median Sales by Month")
```

```
## Warning: Removed 62 rows containing non-finite values (stat_summary).
```



```
garfs %>%
    group_by(month) %>%
    summarise(median(LittleNumber, na.rm = TRUE), mean(LittleNumber, na.rm = TRUE))
```

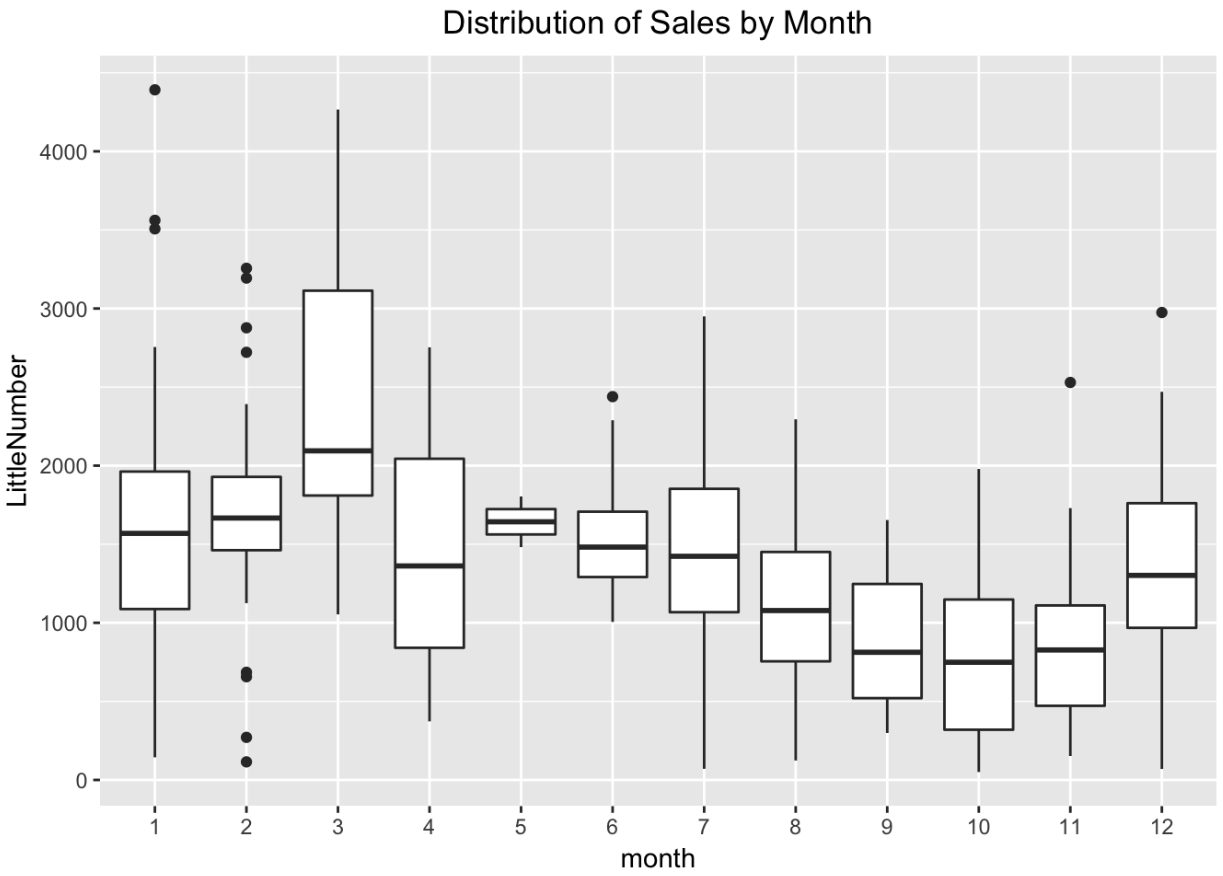
month <fctr>	median(LittleNumber, na.rm = TRUE) <dbl>	mean(LittleNumber, na.rm = TRUE) <dbl>
1	1569.000	1674.7409
2	1666.750	1705.6481
3	2094.250	2430.4334
4	1361.750	1412.0893
5	1642.750	1642.7500
6	1481.750	1557.6094
7	1423.250	1458.6357
8	1078.000	1119.9167

month <fctr>	median(LittleNumber, na.rm = TRUE) <dbl>	mean(LittleNumber, na.rm = TRUE) <dbl>
9	812.500	871.0444
10	749.010	760.0678

1-10 of 12 rows Previous **1** 2 Next

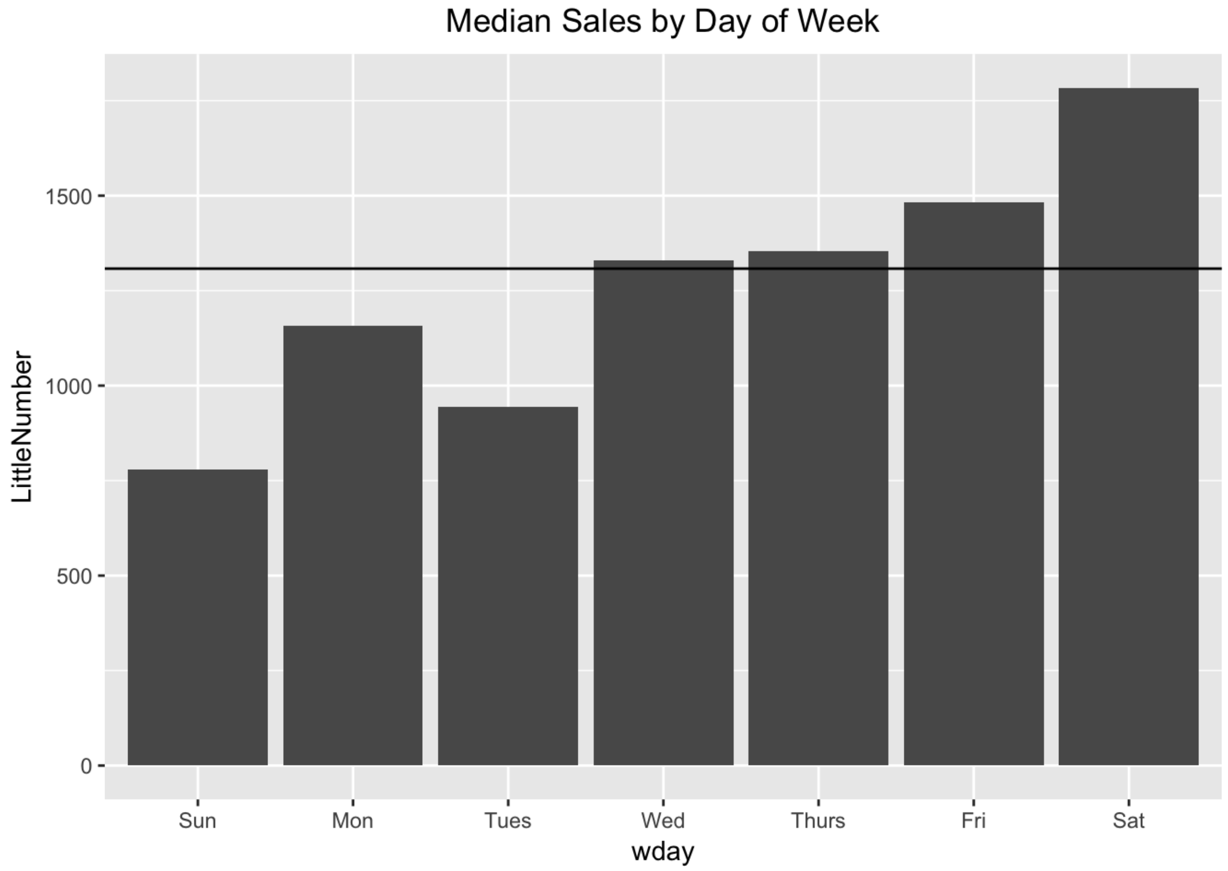
```
garfs %>%
    ggplot(aes(month, LittleNumber)) +
    geom_boxplot() +
#    geom_hline(aes(yintercept = median(LittleNumber, na.rm = TRUE))) +
    ggtitle("Distribution of Sales by Month")
```

```
## Warning: Removed 62 rows containing non-finite values (stat_boxplot).
```



```
garfs %>%
    ggplot(aes(wday, LittleNumber)) +
    stat_summary(fun.y = median, geom = "bar") +
    geom_hline(aes(yintercept = median(LittleNumber, na.rm = TRUE))) +
    ggtitle("Median Sales by Day of Week")
```

```
## Warning: Removed 62 rows containing non-finite values (stat_summary).
```



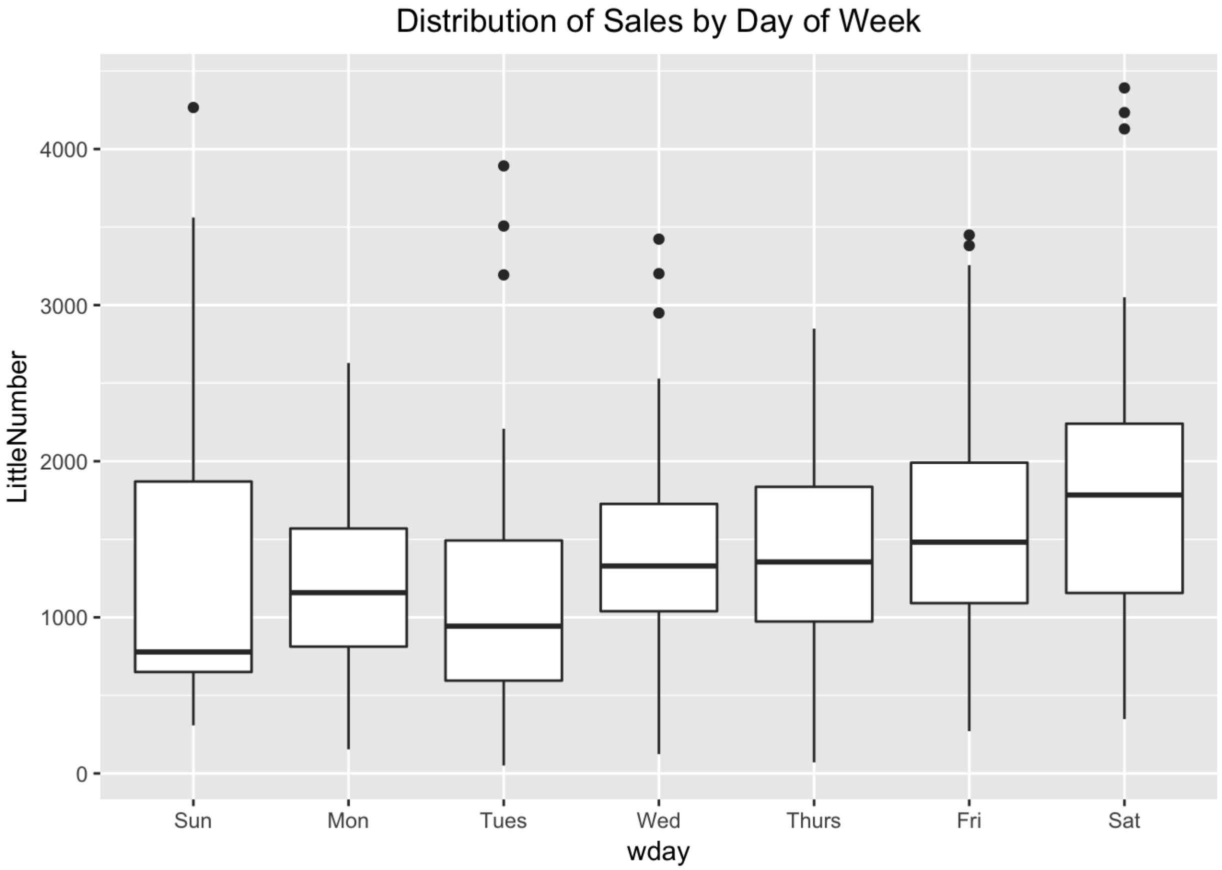
```
garfs %>%
    group_by(wday) %>%
    summarise(median(LittleNumber, na.rm = TRUE), mean(LittleNumber, na.rm = TRUE))
```

wday <ord>	median(LittleNumber, na.rm = TRUE) <dbl>	mean(LittleNumber, na.rm = TRUE) <dbl>
Sun	778.125	1500.028
Mon	1158.000	1201.741
Tues	943.500	1137.866
Wed	1328.965	1396.162
Thurs	1354.750	1332.932
Fri	1481.500	1601.445
Sat	1783.625	1836.017

7 rows

```
garfs %>%
    ggplot(aes(wday, LittleNumber)) +
    geom_boxplot() +
#    geom_hline(aes(yintercept = median(LittleNumber, na.rm = TRUE))) +
    ggtitle("Distribution of Sales by Day of Week")
```

```
## Warning: Removed 62 rows containing non-finite values (stat_boxplot).
```



When identifying outliers, I will arbitrarily call anything lower than **500** or higher than **2000** an outlier.

```
garfs %>%
    filter(LittleNumber < 500)
```

Date <date>	BigNumber <dbl>	LittleNumber <dbl>	NegDrop <dbl>	Dinner <int>	Shift <chr>	CutTime <int>	Made <int>
2017-07-06	438.27	399.25	NA	0	pm	7	81
2017-07-10	379.48	369.50	NA	0	pm	NA	48
2017-07-13	55.30	71.00	-11.00	0	pm	5	6
2017-08-20	253.72	347.50	-41.77	1	pm	NA	76
2017-08-23	158.28	124.00	NA	0	pm	6	24
2017-09-04	637.17	496.50	-67.03	0	pm	7	92
2017-10-01	333.95	308.03	NA	0	pm	6	44
2017-10-02	532.38	499.52	NA	1	pm	NA	83
2017-10-09	427.24	331.17	-79.00	0	pm	NA	81
2017-11-14	408.58	420.75	-22.85	0	pm	7	74

1-10 of 36 rows | 1-8 of 15 columns Previous **1** 2 3 4 Next

```
garfs %>%
    filter(LittleNumber > 2000)
```

Date	BigNumber	LittleNumber	NegDrop	Dinner	Shift	CutTime	Made
<date>	<dbl>	<dbl>	<dbl>	<int>	<chr>	<int>	<int>
2018-01-02	3775.63	3506.75	-278.64	1	dub	NA	523
2018-01-05	2968.32	2755.00	NA	0	dub	NA	413
2018-01-12	2591.36	2417.50	NA	0	dub	NA	334
2018-01-21	3735.21	3561.25	-34.66	0	dub	NA	573
2018-01-31	2280.25	2176.25	-109.70	0	pm	NA	463
2018-02-09	3529.10	3256.00	NA	0	dub	NA	544
2018-02-23	2328.58	2154.00	NA	1	pm	NA	469
2018-02-27	3425.54	3193.75	NA	0	dub	NA	554
2018-03-02	3709.34	3449.50	NA	0	swing	NA	622
2018-03-03	4559.27	4233.50	-99.53	0	swing	NA	675

1-10 of 57 rows | 1-8 of 15 columns Previous **1** 2 3 4 5 6 Next

Cuts

I recognize that getting cut is one of the primary reasons I have bad shifts, and that it is also one of the biggest things holding back my average take-home.

To begin, here is a table breaking down when I got cut (if at all) and how much I made on average.

```
garfs %>%
    filter(Shift == "pm") %>%
    group_by(CutTime) %>%
    summarise(n(), mean(Made))
```

CutTime	n()	mean(Made)
<int>	<int>	<dbl>
5	2	6.50000
6	9	29.11111
7	13	59.30769
8	14	144.00000
9	5	131.80000
NA	219	220.58447

6 rows

A look at the average impact of getting cut:

```
(mean_all <- garfs %>%
    summarise(`Mean of all shifts` = mean(Made)))
```

Mean of all shifts <dbl>
241.4258

1 row

```
(mean_no_cuts <- garfs %>%
    filter(is.na(CutTime)) %>%
    summarise(`Mean of full shifts` = mean(Made)))
```

Mean of full shifts <dbl>
259.7299

1 row

```
(mean_cuts <- garfs %>%
    filter(!is.na(CutTime)) %>%
    summarise(`Mean of cuts` = mean(Made), n()))
```

Mean of cuts <dbl>	n() <int>
134.0189	53

1 row

Miscellaneous

I work mostly nights, but here’s a look at how things break down between days, nights, and doubles:

```
garfs %>%
    group_by(Shift) %>%
    summarise(count = n(), percent_n = round((count / nrow(garfs) * 100), digits = 1), mean(Made), median
(Made))
```

Shift <chr>	count <int>	percent_n <dbl>	mean(Made) <dbl>	median(Made) <dbl>
am	29	8.0	309.3103	302.0
am bar	1	0.3	331.0000	331.0
apres	1	0.3	117.0000	117.0
day swing	1	0.3	339.0000	339.0
dub	9	2.5	509.0000	544.0
pm	262	72.0	198.5840	189.0
pm bar	43	11.8	309.0930	308.0
swing	18	4.9	456.7222	439.5

8 rows

Percentage of time I bought dinner while working:

```
garfs %>%
    summarise(mean(Dinner))
```

mean(Dinner)
<dbl>
NA

1 row

Total made since June 8, 2017:

```
garfs %>%
    summarise(sum(Made))
```

sum(Made)
<int>
87879

1 row

Things to add:

- Add headers to make slides
- Break down by season (and also year)