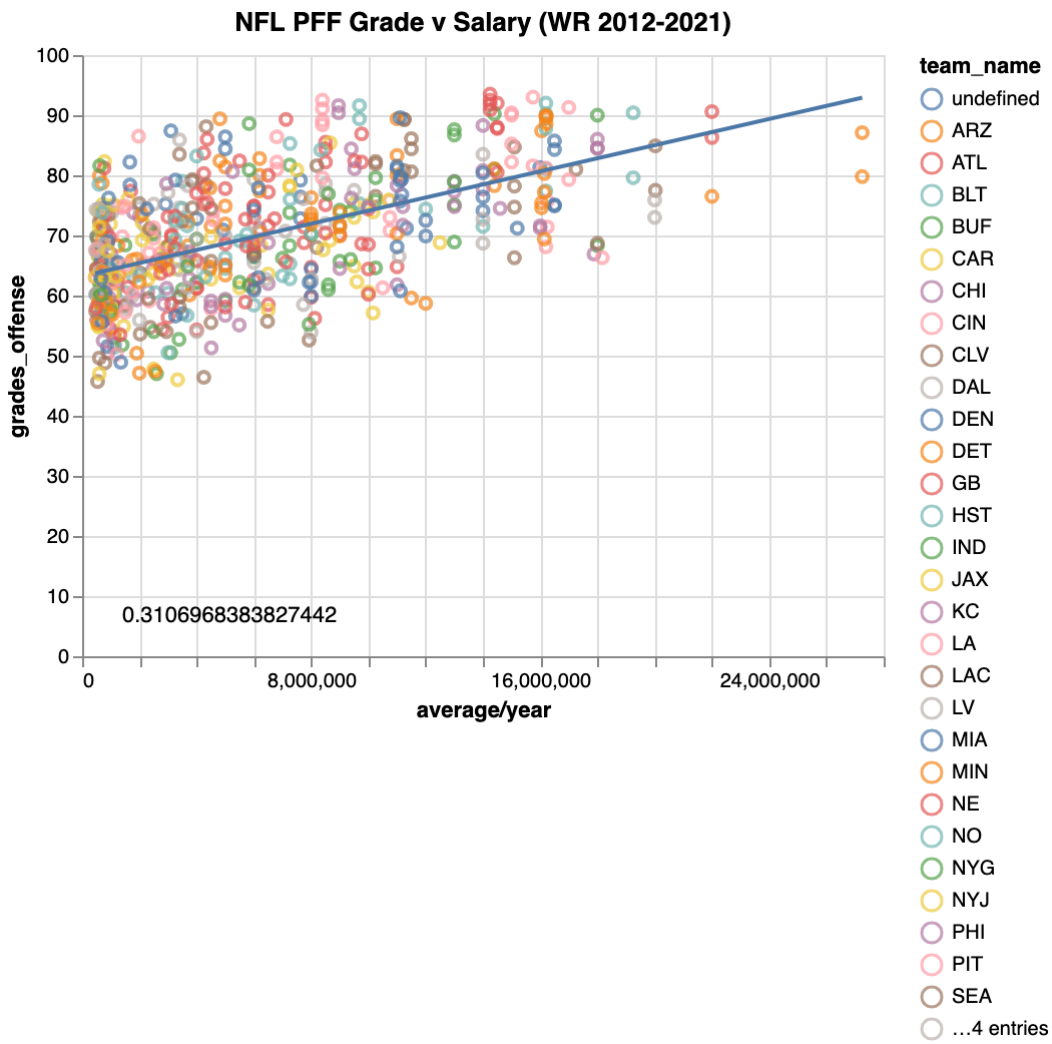
In [ ]:
```
import seaborn as sns
import matplotlib.pyplot as plt
import altair as alt
from vega_datasets import data
from altair import datum
```

In [260…
```
import pandas as pd
rec = pd.read_csv("to_visualize/wr_sal2.csv")
```

In [290…
```
chart = alt.Chart(rec).mark_point().encode(
 x=alt.X("average/year:Q", scale=alt.Scale(domain=[0, 28000000])),
 y=alt.Y("grades_offense:Q", scale=alt.Scale(domain=[0,100])),
 color="team_name:N",
 tooltip="player_name:N",
).properties(
 title={
 "text": "NFL PFF Grade v Salary (WR 2012-2021)"
 }
)
line = chart.transform_regression('average/year', 'grades_offense').mark_line()
params = alt.Chart(rec).transform_regression(
    'average/year', 'grades_offense', params=True
).mark_text(align='left').encode(
    x=alt.value(20),
    y=alt.value(280),
    text='rSquared:N'
)
chart+line+params
```

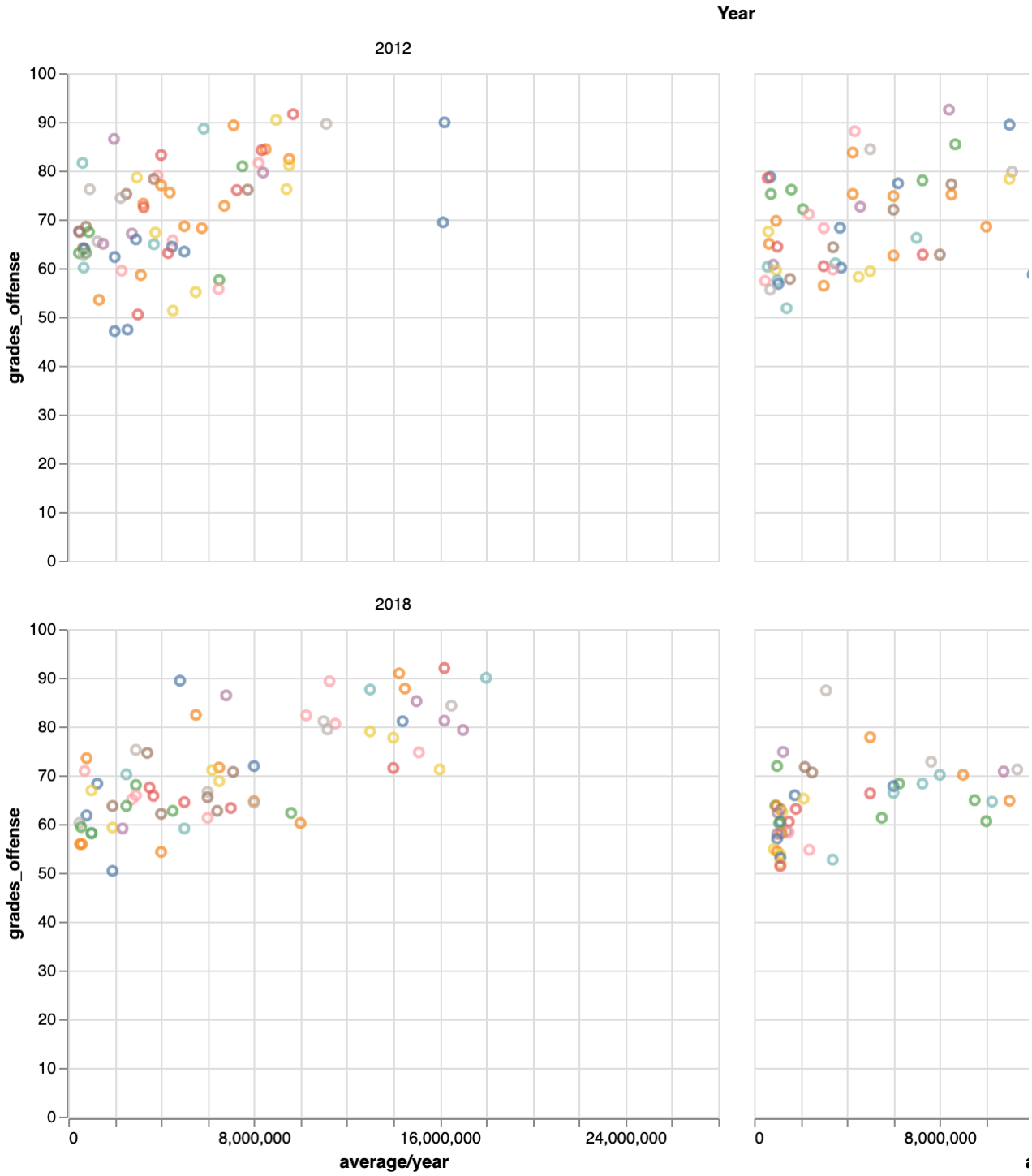
Out[290…



In [286…

```
chart = alt.Chart(rec).mark_point().encode(
 x=alt.X("average/year:Q", scale=alt.Scale(domain=[0, 28000000])),
 y=alt.Y("grades_offense:Q", scale=alt.Scale(domain=[0,100])),
 color="team_name:N",
 tooltip="player_name:N",
).properties(
 title={
 "text": "NFL PFF Grade v Salary"
 }
).transform_filter(
    (alt.datum.Year == 2012) | (alt.datum.Year == 2015) | (alt.datum.Year == 201
)
# line = chart.transform_regression('average/year', 'grades_offense').mark_line(
# params = alt.Chart(end).transform_regression(
#     'win_percentage', 'avg_diff_wr', params=True
# ).mark_text(align='left').encode(
#     x=alt.value(20),
#     y=alt.value(280),
#     text='rSquared:N'
# )
multichart = chart.facet("Year", columns=2)
multichart
```

Out [286…

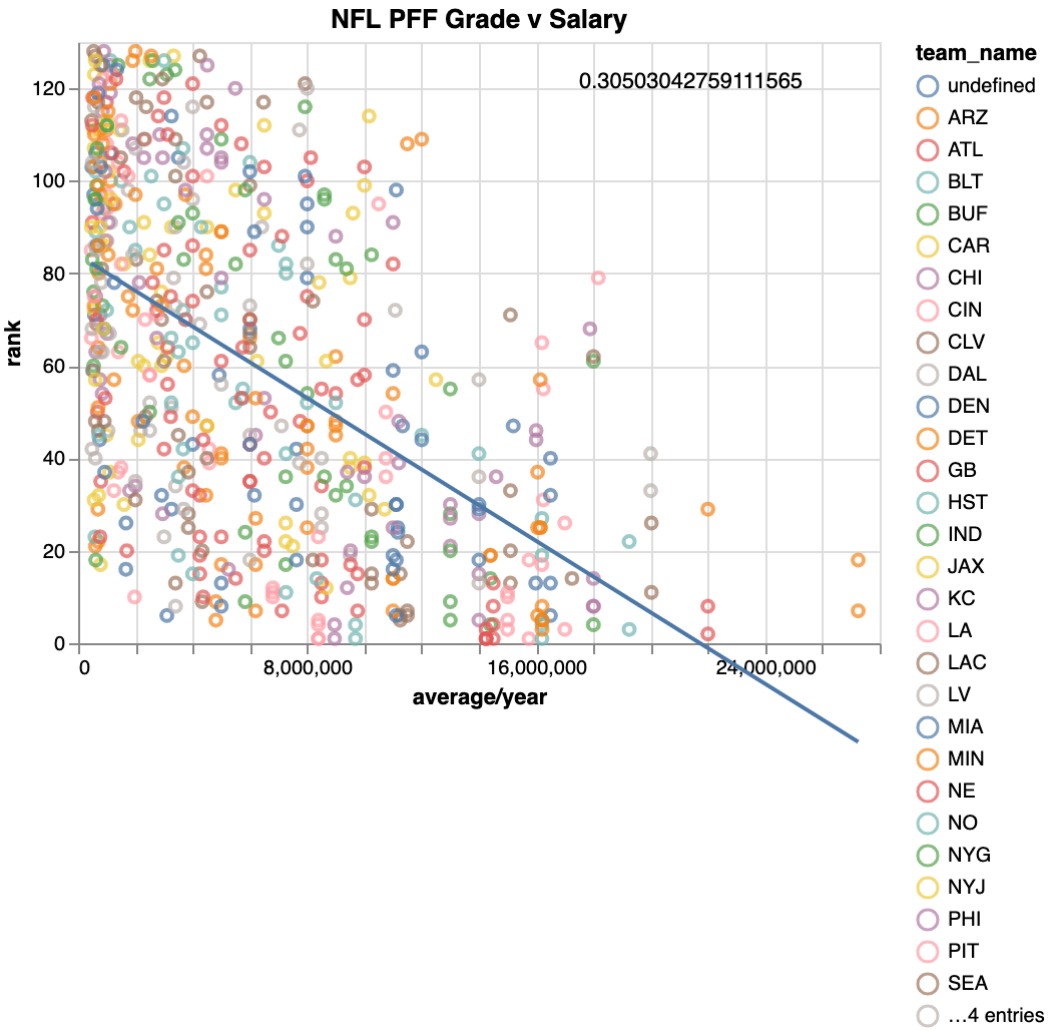


In [293…

```
chart = alt.Chart(rec).mark_point().encode(
 x=alt.X("average/year:Q", scale=alt.Scale(domain=[0, 28000000])),
 y=alt.Y("rank:Q", scale=alt.Scale(domain=[0,128])),
 color="team_name:N",
 tooltip="player_name:N",
).properties(
 title={
 "text": "NFL PFF Grade v Salary"
 }
)
line = chart.transform_regression('average/year', 'rank').mark_line()
params = alt.Chart(rec).transform_regression(
    'average/year', 'rank', params=True
```

```
).mark_text(align='left').encode(
    x=alt.value(250),
    y=alt.value(20),
    text='rSquared:N'
)
chart+line+params
```

Out [293…

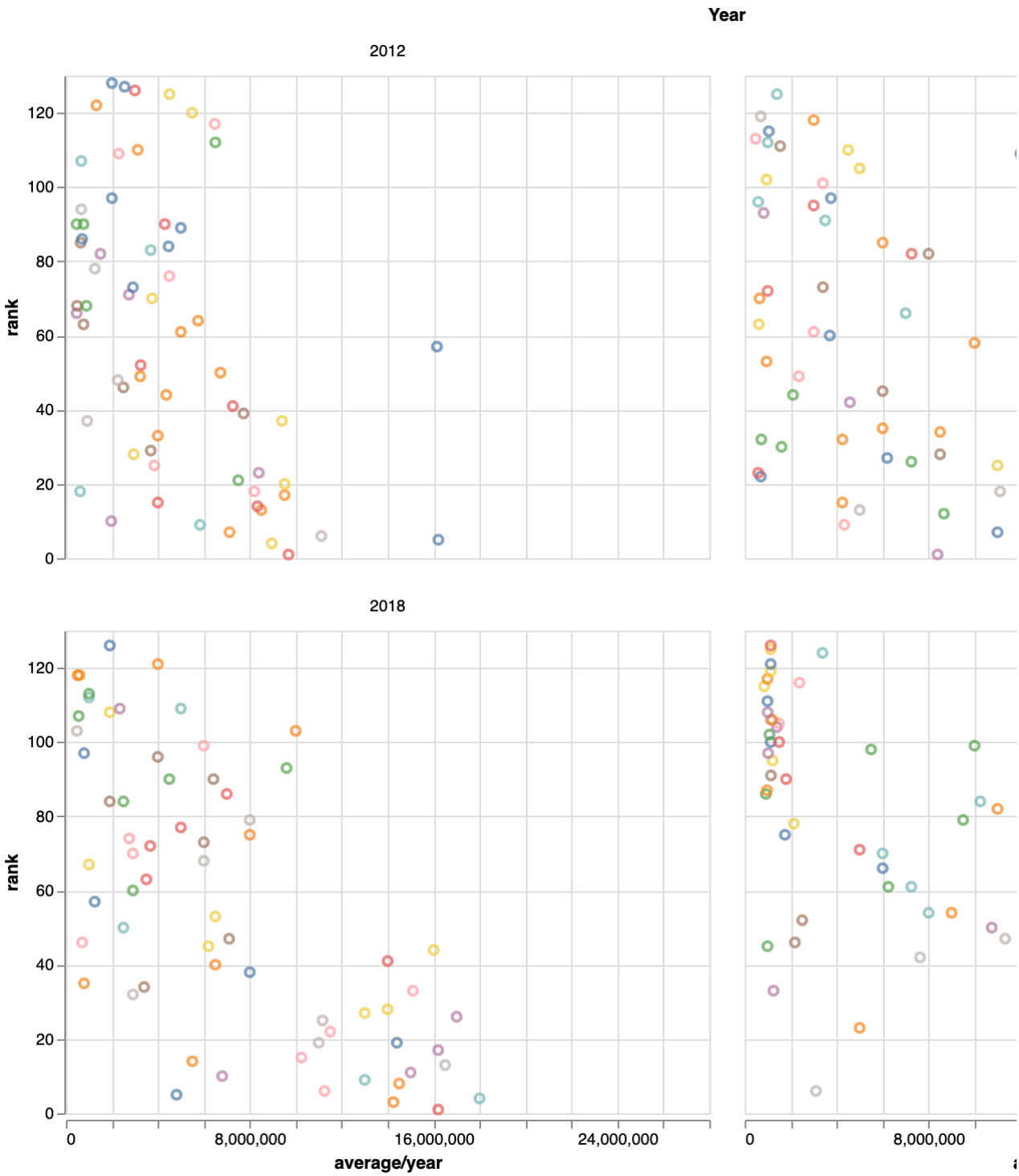


In [294…

```
chart = alt.Chart(rec).mark_point().encode(
 x=alt.X("average/year:Q", scale=alt.Scale(domain=[0, 28000000])),
 y=alt.Y("rank:Q", scale=alt.Scale(domain=[0,128])),
 color="team_name:N",
 tooltip="player_name:N",
).properties(
 title={
 "text": "NFL PFF Grade v Salary"
 }
).transform_filter(
    (alt.datum.Year == 2012) | (alt.datum.Year == 2015) | (alt.datum.Year == 201
)
# line = chart.transform_regression('average/year', 'grades_offense').mark_line(
# params = alt.Chart(end).transform_regression(
#     'win_percentage', 'avg_diff_wr', params=True
# ).mark_text(align='left').encode(
#     x=alt.value(20),
#     y=alt.value(280),
```

```
#     text='rSquared:N'
# )
multichart = chart.facet("Year", columns=2)
multichart
```

Out [294…

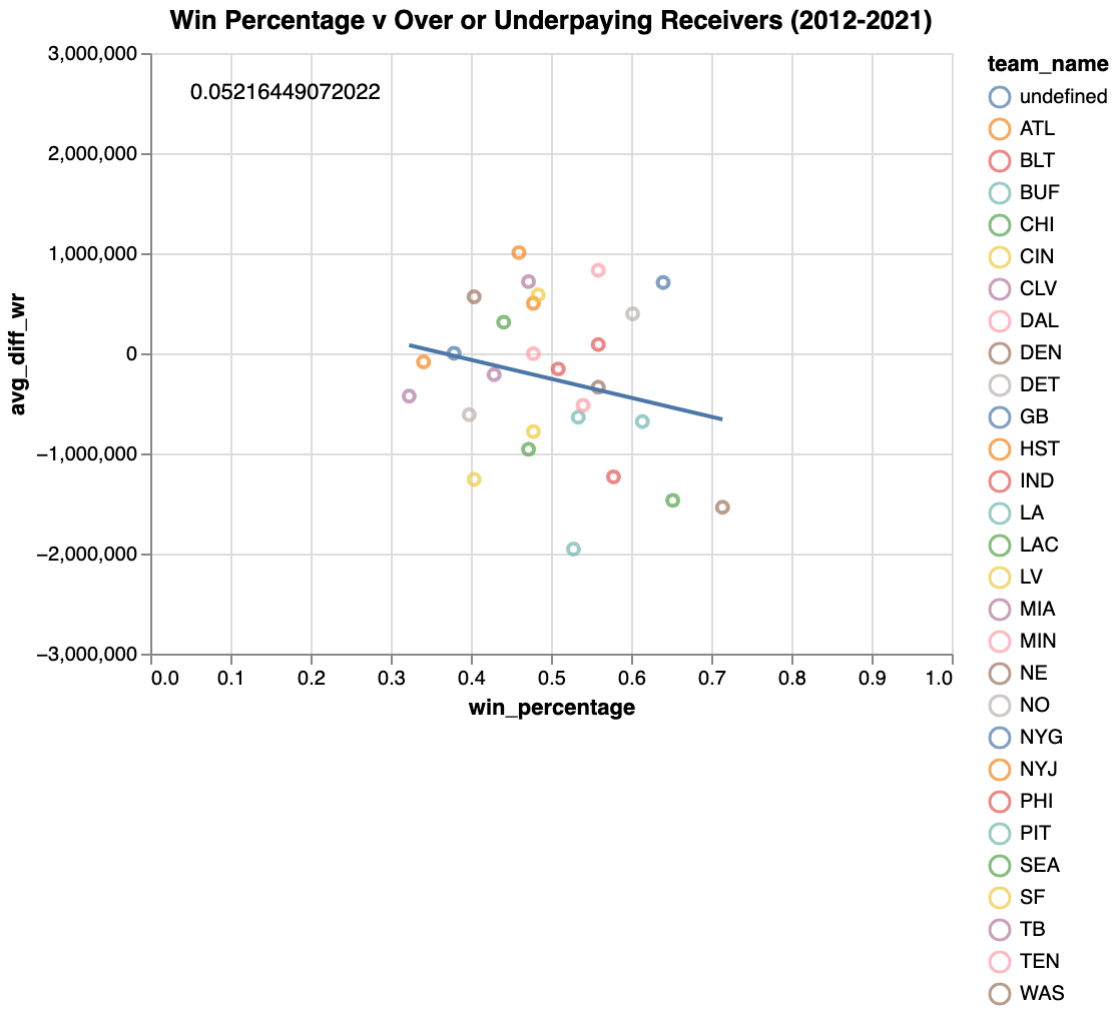


In [296…

```
end = pd.read_csv("final/sample_end.csv")
chart = alt.Chart(end).mark_point().encode(
 x=alt.X("win_percentage:Q", scale=alt.Scale(domain=[0, 1])),
 y=alt.Y("avg_diff_wr:Q", scale=alt.Scale(domain=[-3000000,3000000])),
 color="team_name:N",
 tooltip="team_name:N",
).properties(
```

```
 title={
 "text": "Win Percentage v Over or Underpaying Receivers (2012-2021)"
 }
)
line = chart.transform_regression('win_percentage', 'avg_diff_wr').mark_line()
params = alt.Chart(end).transform_regression(
    'win_percentage', 'avg_diff_wr', params=True
).mark_text(align='left').encode(
    x=alt.value(20),
    y=alt.value(20),
    text='rSquared:N'
)
chart+line+params
```

Out [296…



Negative means underpaying, postive means overpaying

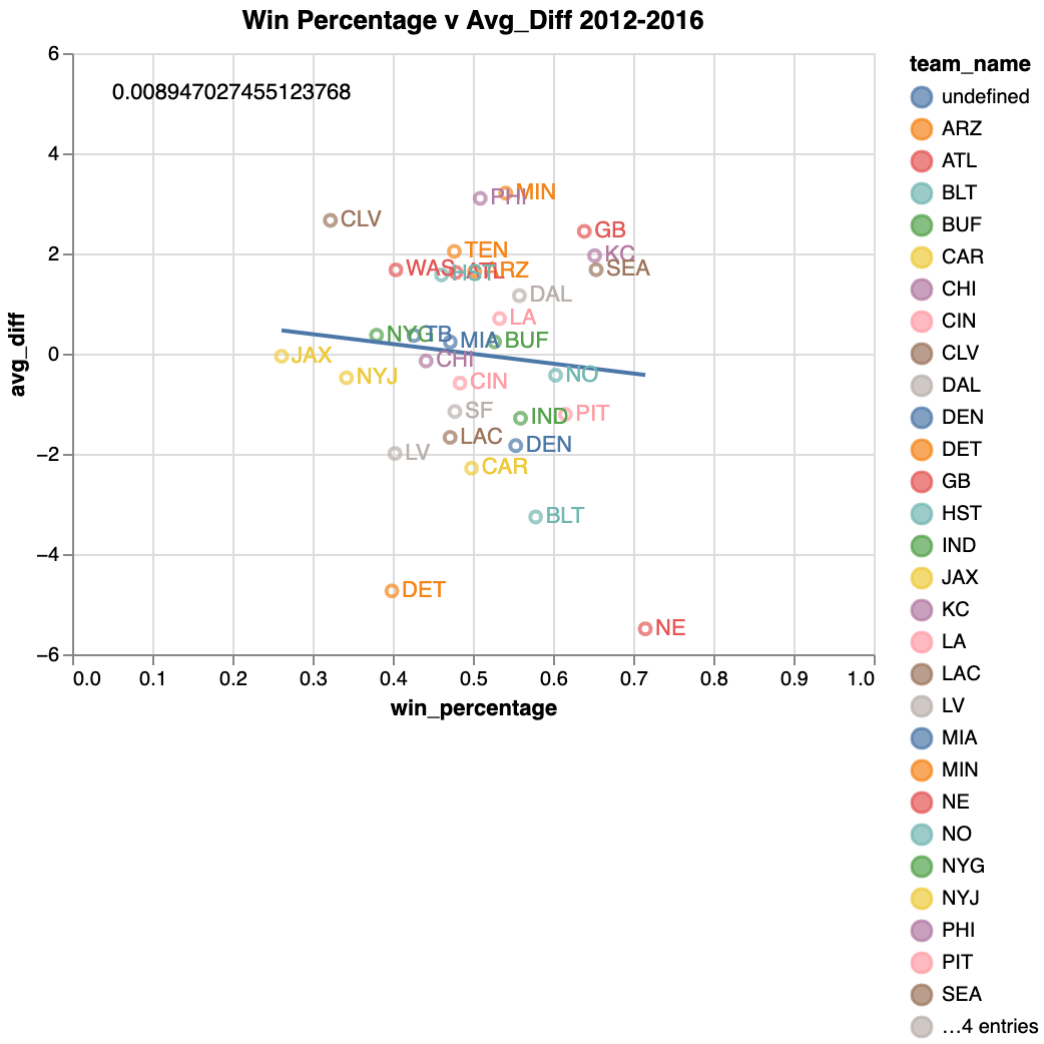
Mean Filling for

In [331…

```
done = pd.read_csv("to_visualize/comparison_fillna.csv")
chart = alt.Chart(done).mark_point().encode(
 x=alt.X("win_percentage:Q", scale=alt.Scale(domain=[0, 1])),
 y=alt.Y("avg_diff:Q", scale=alt.Scale(domain=[-5.5,5.5])),
 color="team_name:N",
 text = "team_name:N",
).properties(
 title={
```

```
 "text": "Win Percentage v Avg_Diff 2012-2021 ()"
 }
)
text = chart.mark_text(
    align='left',
    baseline='middle',
    dx=5
).encode(
    text='team_name'
)
line = chart.transform_regression('win_percentage', 'avg_diff').mark_line()
params = alt.Chart(done).transform_regression(
    'win_percentage', 'avg_diff', params=True
).mark_text(align='left').encode(
    x=alt.value(20),
    y=alt.value(20),
    text='rSquared:N'
)
chart+line+params+text
```

Out [331…



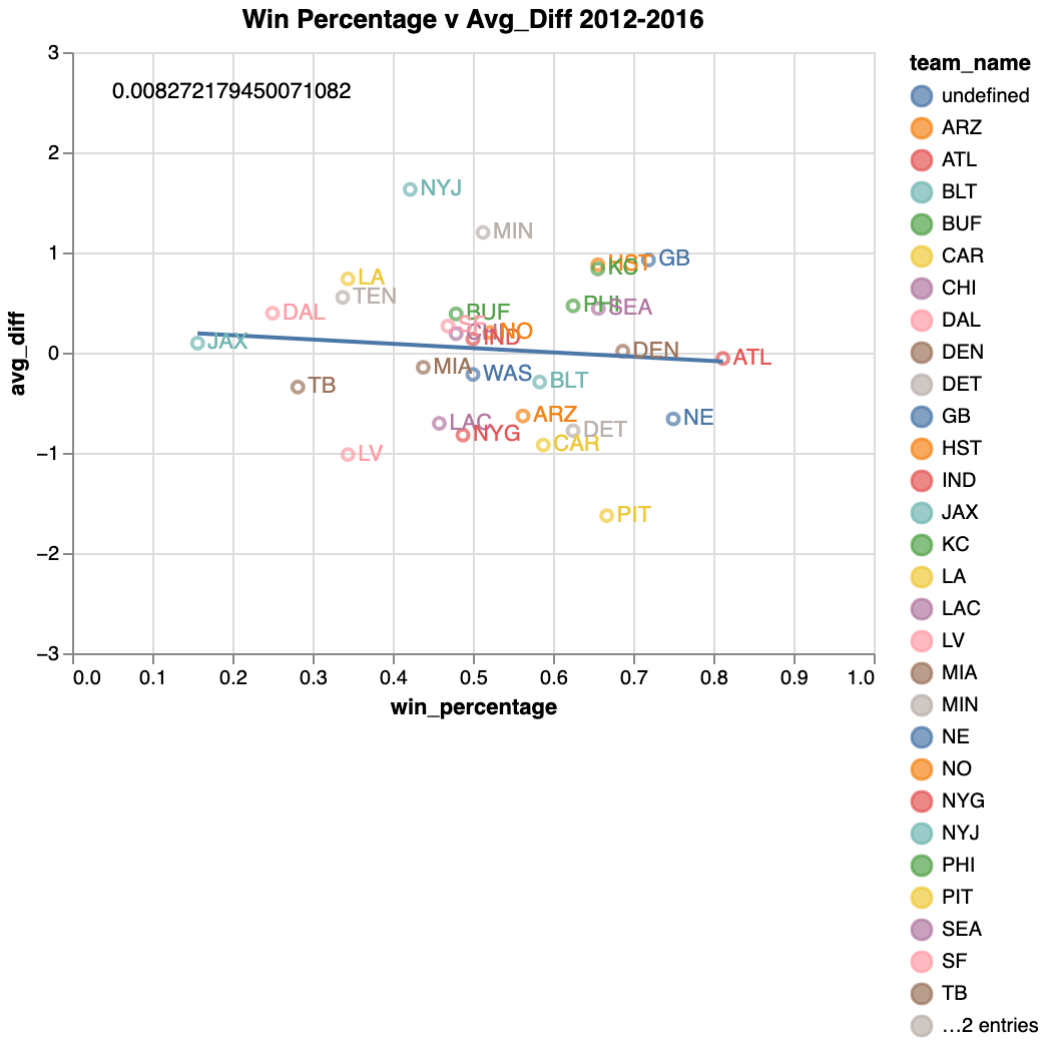
Listwise Deletion

In [329…

```
done = pd.read_csv("to_visualize/comparison12_16.csv")
chart = alt.Chart(done).mark_point().encode(
 x=alt.X("win_percentage:Q", scale=alt.Scale(domain=[0, 1])),
```

```
 y=alt.Y("avg_diff:Q", scale=alt.Scale(domain=[-3,3])),
 color="team_name:N",
 text = "team_name:N",
).properties(
 title={
 "text": "Win Percentage v Avg_Diff 2012-2016"
 }
)
text = chart.mark_text(
    align='left',
    baseline='middle',
    dx=5
).encode(
    text='team_name'
)
line = chart.transform_regression('win_percentage', 'avg_diff').mark_line()
params = alt.Chart(done).transform_regression(
    'win_percentage', 'avg_diff', params=True
).mark_text(align='left').encode(
    x=alt.value(20),
    y=alt.value(20),
    text='rSquared:N'
)
chart+line+params+text
```

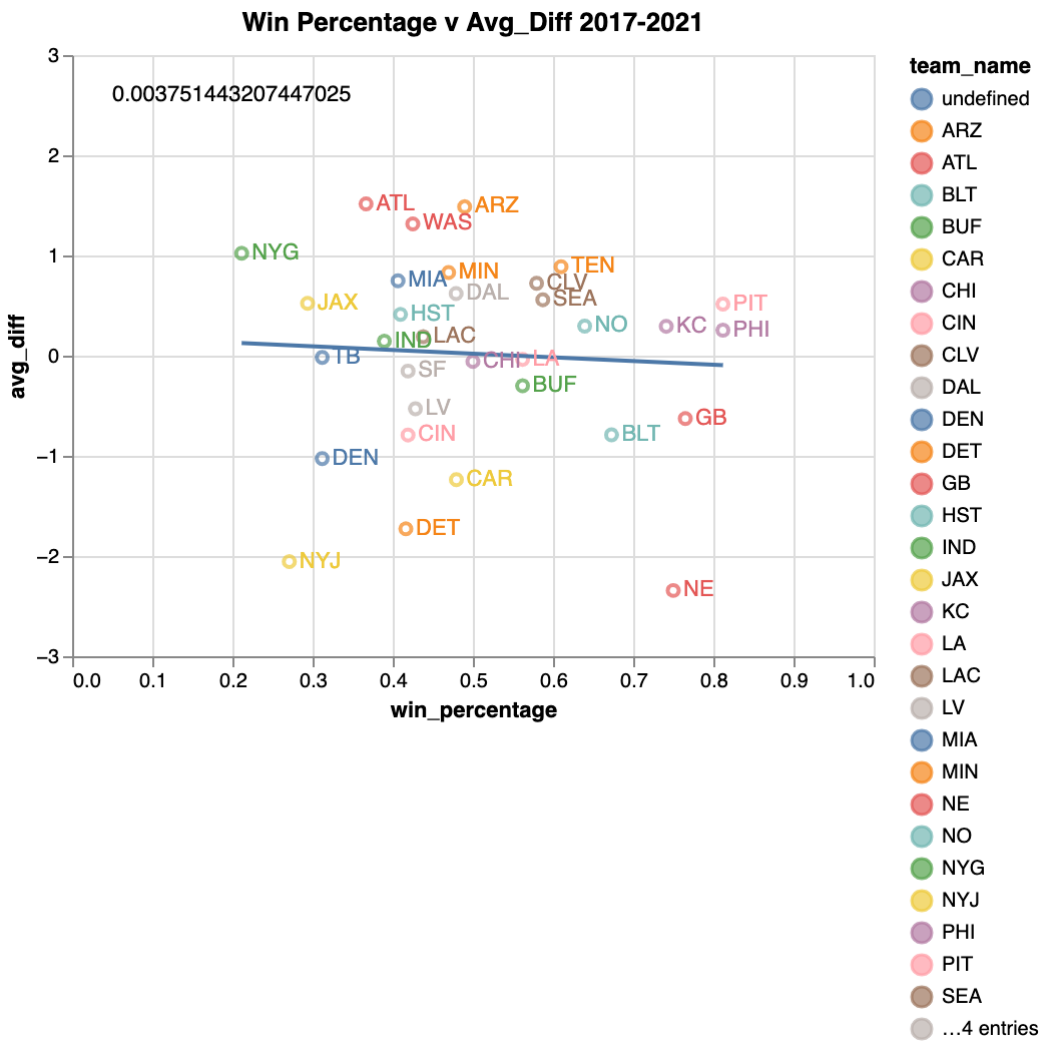
Out [329…



In [303…

```
done = pd.read_csv("to_visualize/comparison17_21.csv")
chart = alt.Chart(done).mark_point().encode(
 x=alt.X("win_percentage:Q", scale=alt.Scale(domain=[0, 1])),
 y=alt.Y("avg_diff:Q", scale=alt.Scale(domain=[-3,3])),
 color="team_name:N",
 # tooltip="team_name:N",
 text = "team_name:N",
).properties(
 title={
 "text": "Win Percentage v Avg_Diff 2017-2021"
 }
)
text = chart.mark_text(
    align='left',
    baseline='middle',
    dx=5
).encode(
    text='team_name'
)
line = chart.transform_regression('win_percentage', 'avg_diff').mark_line()
params = alt.Chart(done).transform_regression(
    'win_percentage', 'avg_diff', params=True
).mark_text(align='left').encode(
    x=alt.value(20),
    y=alt.value(20),
    text='rSquared:N'
)
chart+line+params+text
```

Out[303…

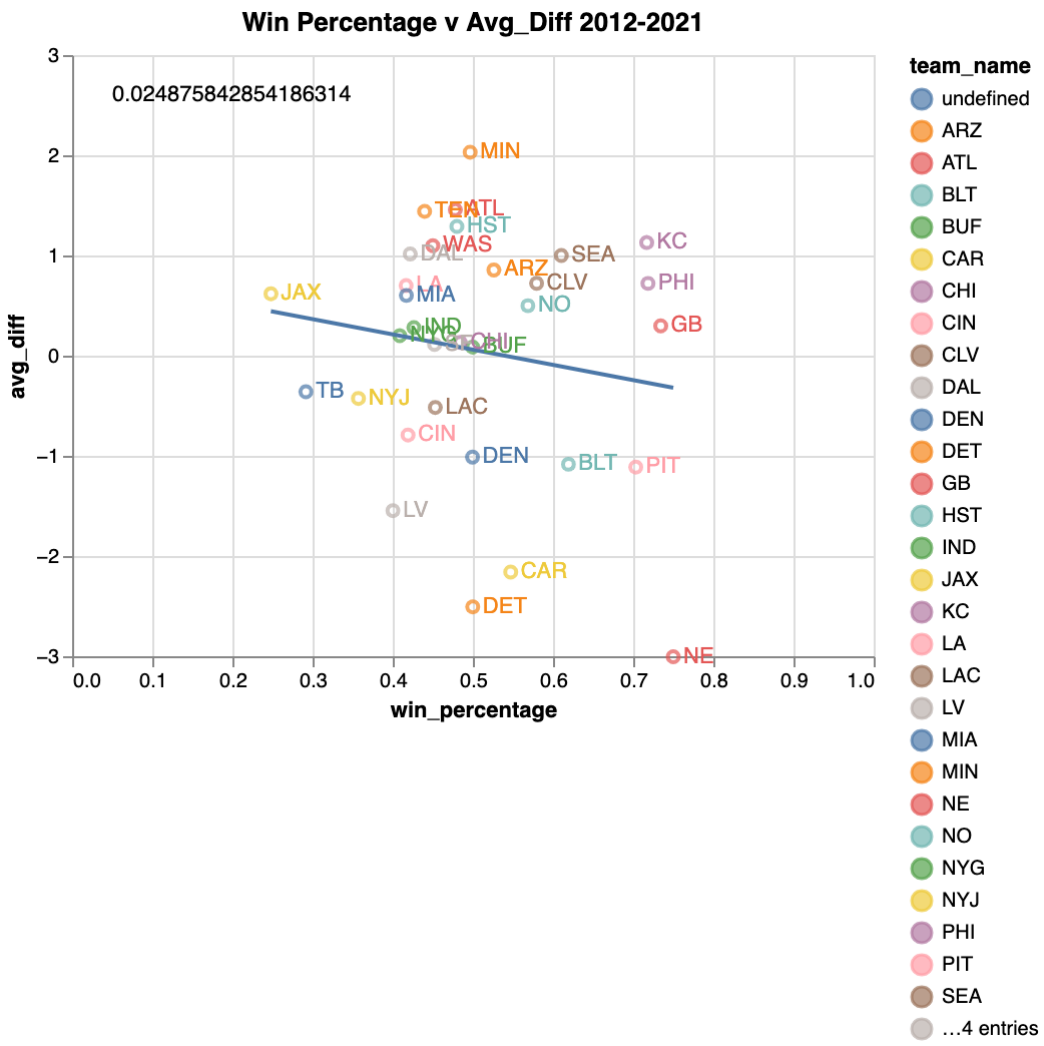


In [305…

```
done = pd.read_csv("to_visualize/comparison.csv")
chart = alt.Chart(done).mark_point().encode(
 x=alt.X("win_percentage:Q", scale=alt.Scale(domain=[0, 1])),
 y=alt.Y("avg_diff:Q", scale=alt.Scale(domain=[-3,3])),
 color="team_name:N",
 # tooltip="team_name:N",
 text = "team_name:N",
).properties(
 title={
 "text": "Win Percentage v Avg_Diff 2012-2021"
 }
)
text = chart.mark_text(
    align='left',
    baseline='middle',
    dx=5
).encode(
    text='team_name'
)
line = chart.transform_regression('win_percentage', 'avg_diff').mark_line()
params = alt.Chart(done).transform_regression(
    'win_percentage', 'avg_diff', params=True
).mark_text(align='left').encode(
    x=alt.value(20),
    y=alt.value(20),
```

```
    text='rSquared:N'
)
chart+line+params+text
```

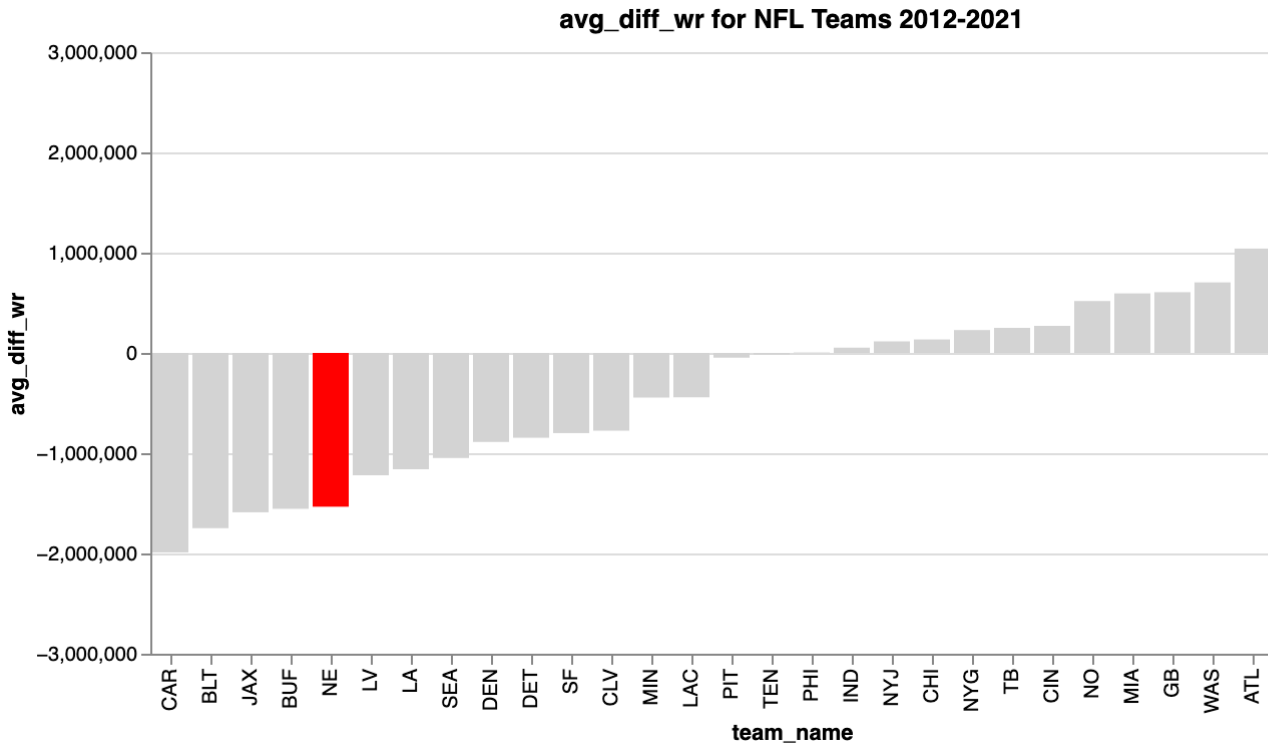
Out[305…



In [324…

```
done = pd.read_csv("to_visualize/wr_underVover.csv")
chart = alt.Chart(done).mark_bar().encode(
 x=alt.X("team_name:N", sort ='y'),
 y=alt.Y("avg_diff_wr:Q", scale=alt.Scale(domain=[-3000000,3000000])),
 # text = "team_name:N",
 color=alt.condition(alt.datum.team_name == 'NE', alt.value('red'), alt.value('l
).properties(
 title={
 "text": "avg_diff_wr for NFL Teams 2012-2021"
 }
)
chart
```

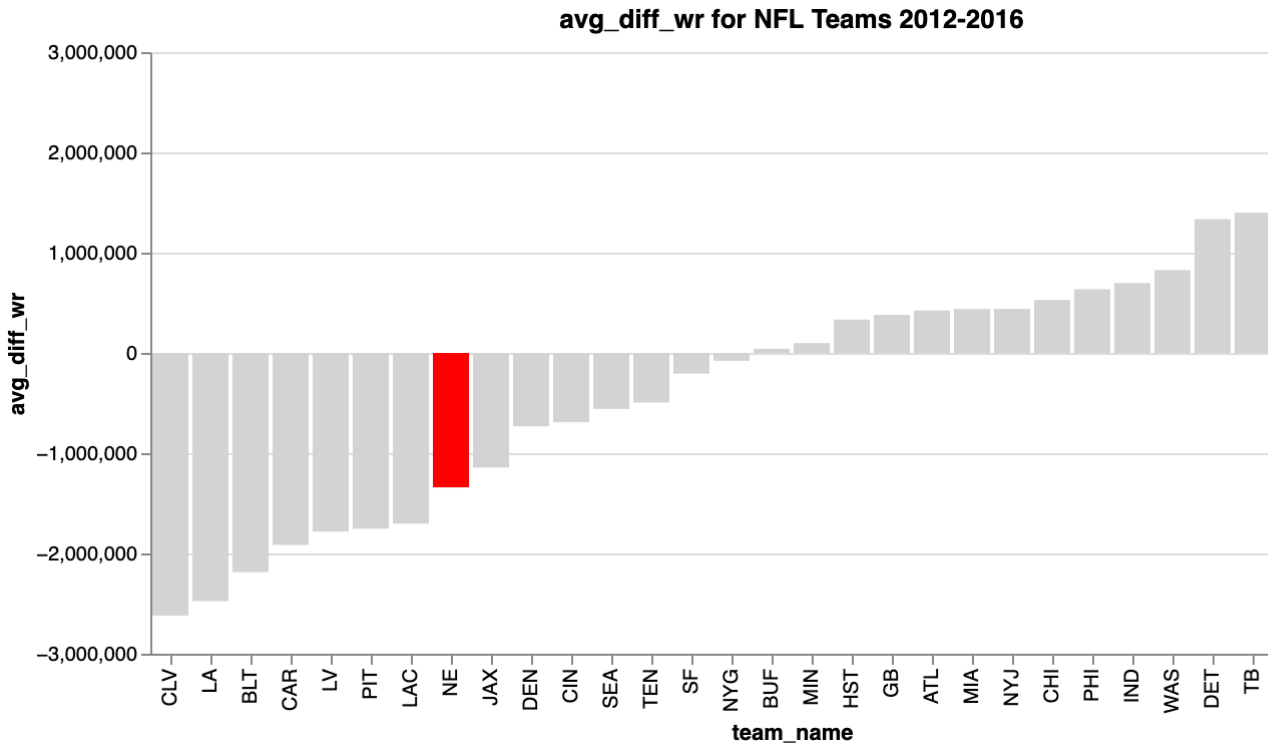
Out[324…



In [325…

```
done = pd.read_csv("to_visualize/wr_underVover12_16.csv")
chart = alt.Chart(done).mark_bar().encode(
 x=alt.X("team_name:N", sort ='y'),
 y=alt.Y("avg_diff_wr:Q", scale=alt.Scale(domain=[-3000000,3000000])),
 # text = "team_name:N",
 color=alt.condition(alt.datum.team_name == 'NE', alt.value('red'), alt.value('l
).properties(
 title={
 "text": "avg_diff_wr for NFL Teams 2012-2016"
 }
)
chart
```

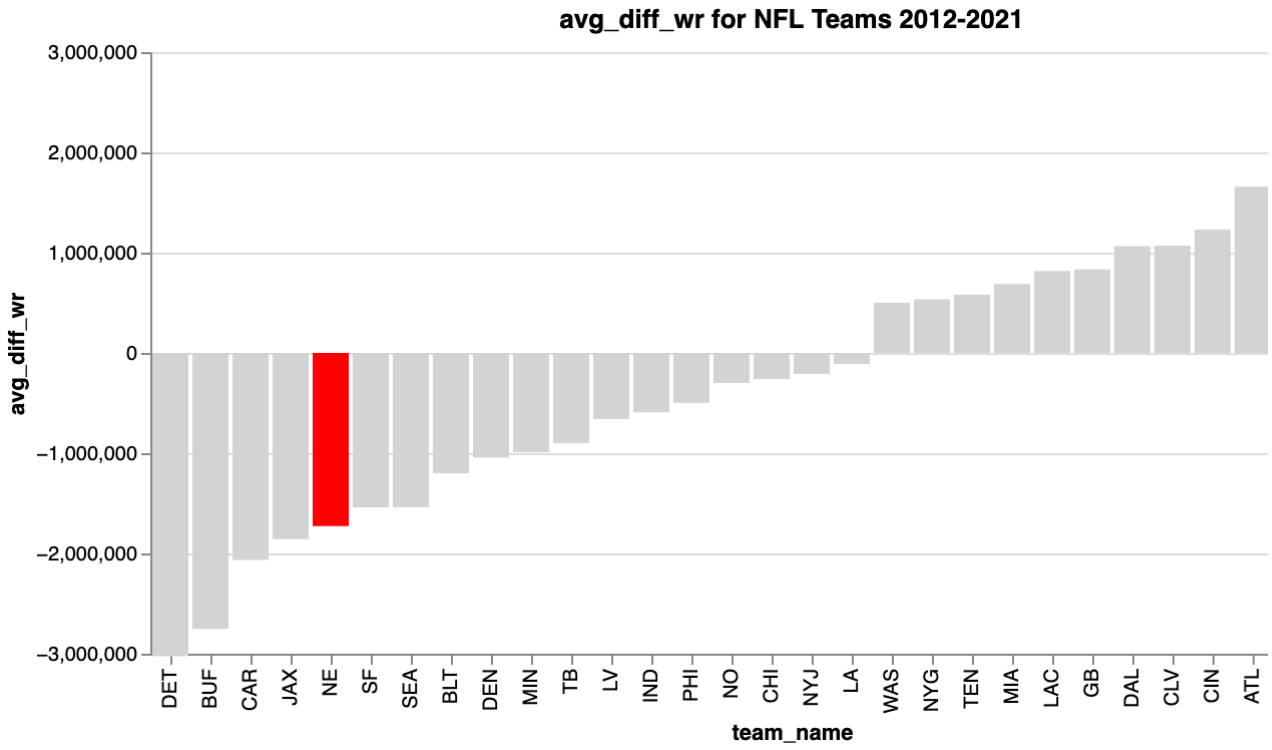
Out [325…



In [326…

```
done = pd.read_csv("to_visualize/wr_underVover17_21.csv")
chart = alt.Chart(done).mark_bar().encode(
 x=alt.X("team_name:N", sort ='y'),
 y=alt.Y("avg_diff_wr:Q", scale=alt.Scale(domain=[-3000000,3000000])),
 # text = "team_name:N",
 color=alt.condition(alt.datum.team_name == 'NE', alt.value('red'), alt.value('l
).properties(
 title={
 "text": "avg_diff_wr for NFL Teams 2012-2021"
 }
)
chart
```

Out[326…



In [ ]:
```
# patriots case study, paying receivers v receiver grade rank, v win percentage,
```