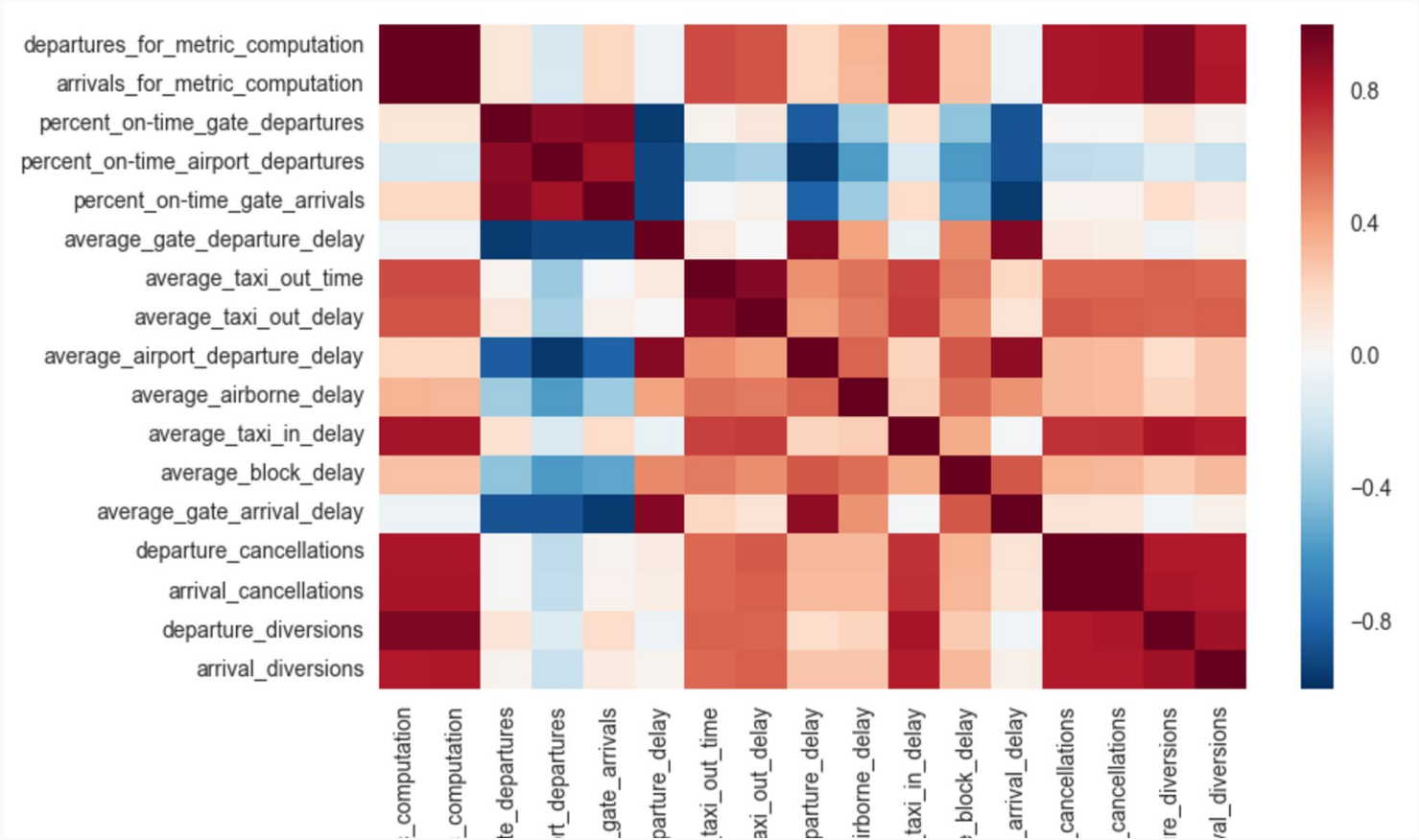
PCA Recap P7

Novel Applications of PCA

By John Marin

EDA

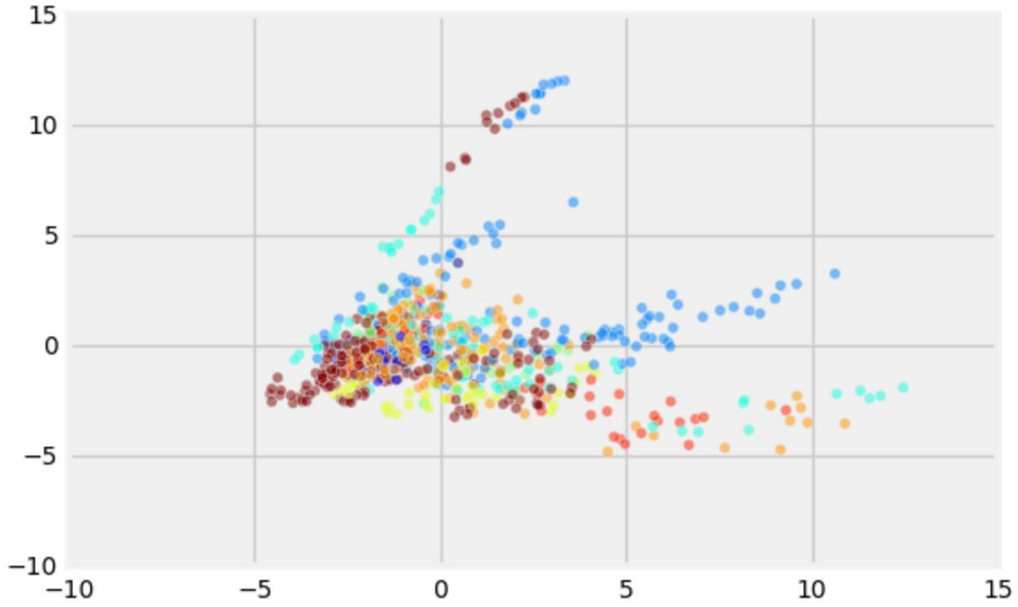

departures_for_metric_computation
arrivals_for_metric_computation
percent_on-time_gate_departures
percent_on-time_airport_departures
percent_on-time_gate_arrivals
average_gate_departure_delay
average_taxi_out_time
average_taxi_out_delay
average_airport_departure_delay
average_airborne_delay
average_taxi_in_delay
average_block_delay
average_gate_arrival_delay
departure_cancellations
arrival_cancellations
departure_diversions
arrival_diversions
0.8
0.4
0.0
−0.4
−0.8

PC1 v. PC2

```
pca = PCA(n_components = 2)
df_plot = pd.DataFrame(pca.fit_transform(all_flights))

plt.scatter(df_plot[0], df_plot[1], c = full_df['faa_le'].values, alpha=0.5)
```

```
<matplotlib.collections.PathCollection at 0x119baef90>
```

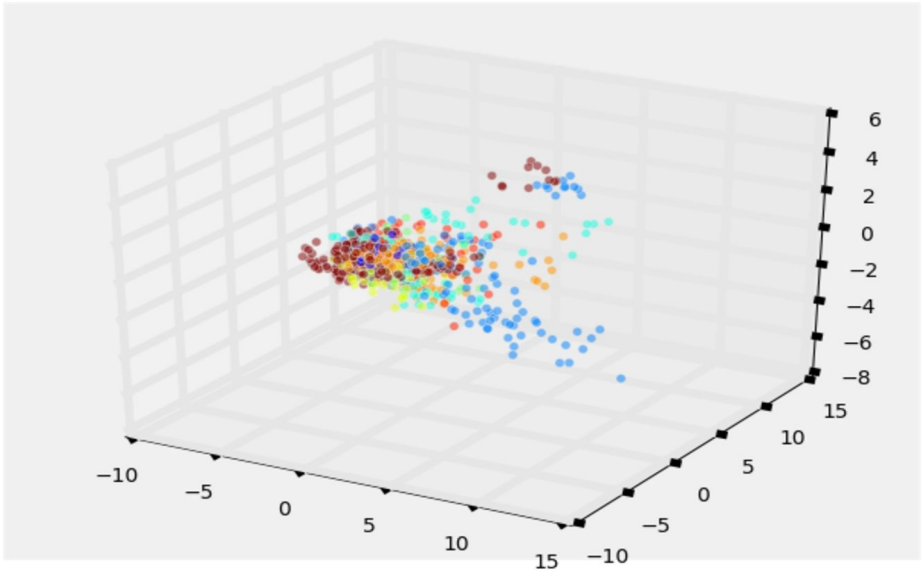


PC1 v. PC2 v. PC3

```
[18]: pca3 = PCA(n_components = 3)
df_plot3 = pd.DataFrame(pca3.fit_transform(all_flights))

#plt.scatter(df_plot3[0], df_plot3[1], c = full_df['faa_le'].values, alpha=0
fig = pylab.figure()
my_axs = Axes3D(fig)
my_axs.scatter(df_plot3[0], df_plot3[1], df_plot3[2],c = full_df['faa_le'].v
```

```
[18]: <mpl_toolkits.mplot3d.art3d.Path3DCollection at 0x116053050>
```

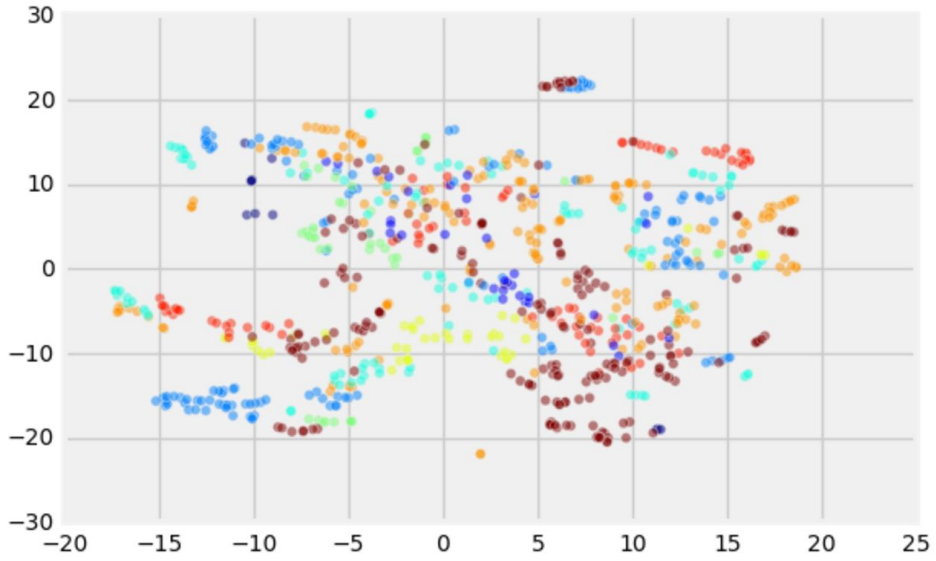


tsne

```
tsnex2 = TSNE(n_components=2)
tsne_X = tsnex2.fit_transform(all_flights)
```

```
plt.scatter(tsne_X[:,0], tsne_X[:,1], c = full_df['faa_le'].values, alpha=0.5)
```

```
<matplotlib.collections.PathCollection at 0x119c29210>
```

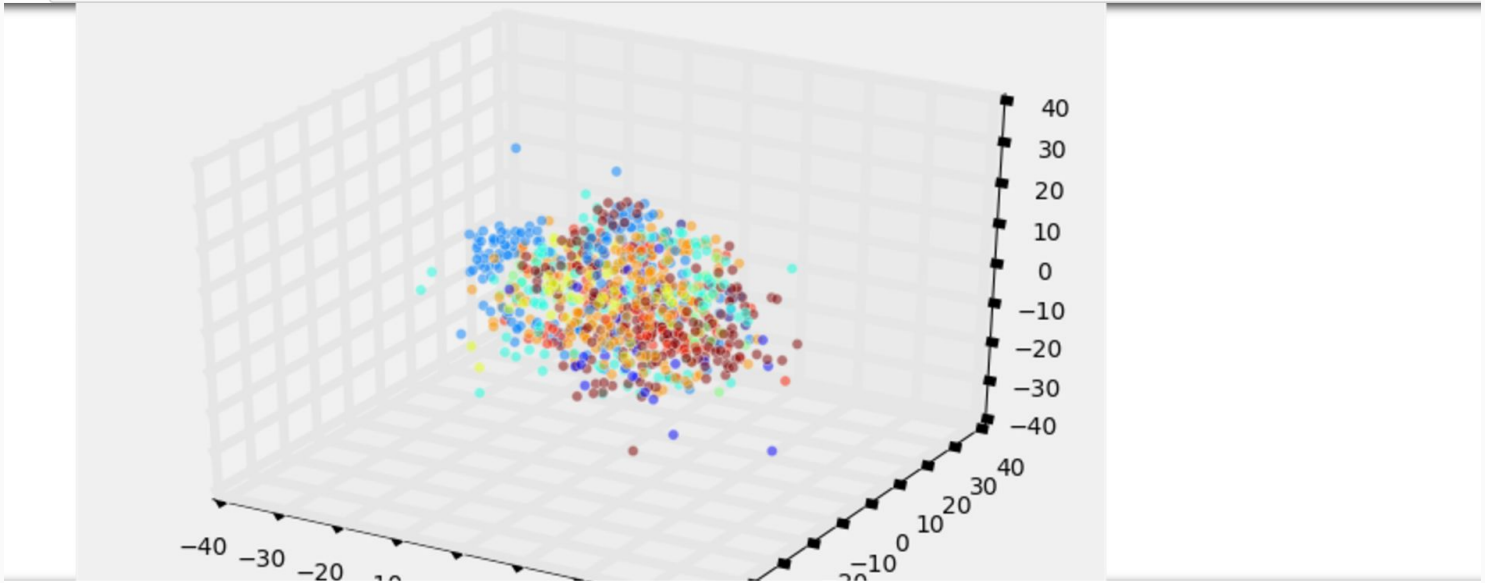


Tsne in 3d - what if we could cluster this and run another PCA?

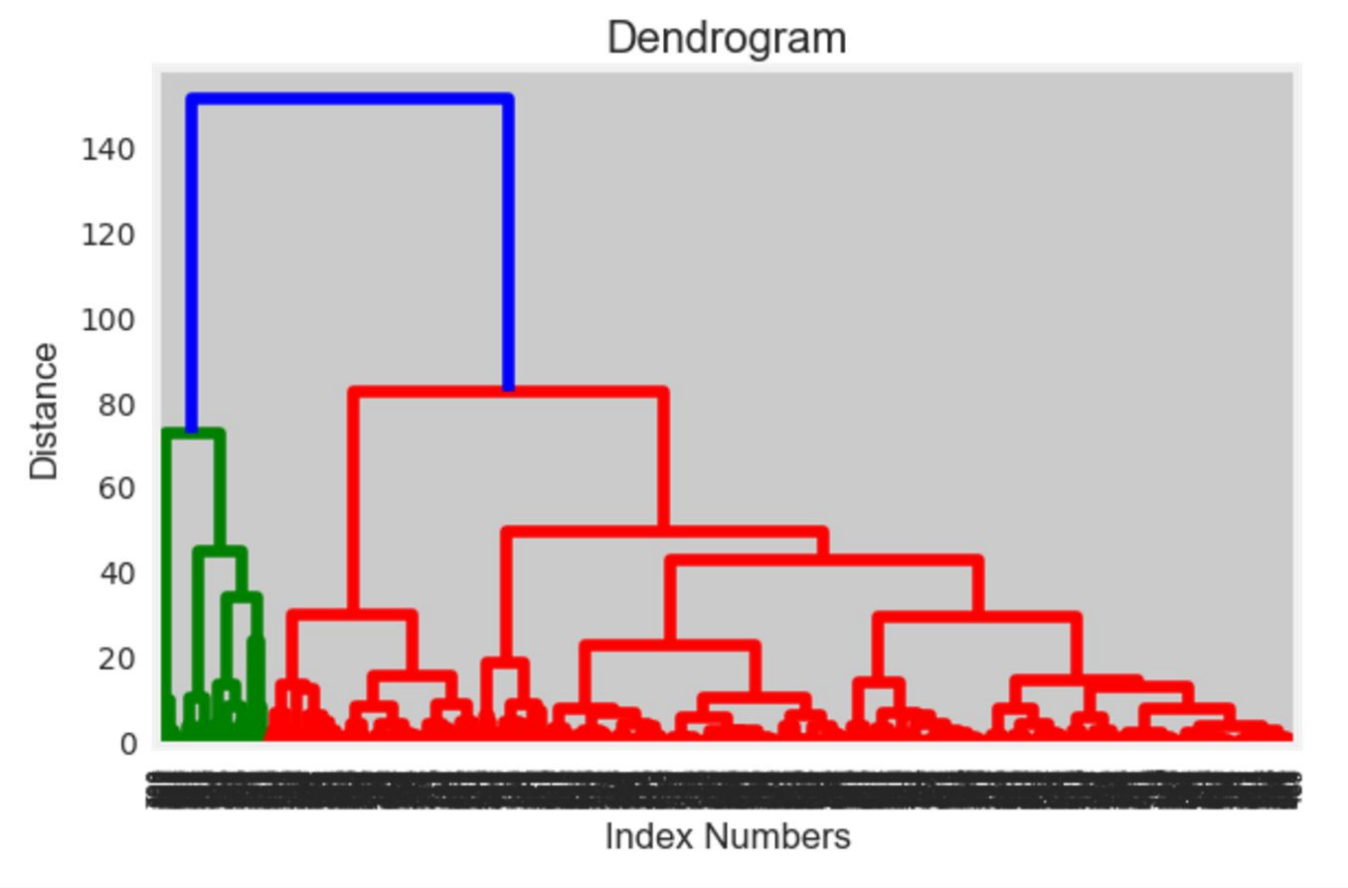
```
]:
tsnex3 = TSNE(n_components=3)
tsne_X3 = tsnex3.fit_transform(all_flights)
```

```
]:
fig = pylab.figure()
my_axs = Axes3D(fig)


my_axs.scatter(tsne_X3[:,0], tsne_X3[:,1], tsne_X3[:,2],c = full_df['faa_le'].values, alpha=0.5)
```



Dendrogram



PCA on the `bad` group does reveal some clustering

```
#run PCA on the bad group...

pca_bf = PCA()
df_plot_af = pd.DataFrame(pca_bf.fit_transform(all_flights[all_flights["g_b_split"]==1]))

plt.scatter(df_plot_af[0], df_plot_af[1],  alpha=0.5)
```

```
<matplotlib.collections.PathCollection at 0x120ef8d90>
```