# Run the Kmeans algorithm and get the index of data points clusters

# source: <https://towardsdatascience.com/k-means-clustering-algorithm-applications-evaluation-methods-and-drawbacks-aa03e644b48a>

sse = []

list\_k = list(range(1, 10))

for k in list\_k:

km = KMeans(n\_clusters=k)

km.fit(X\_train)

sse.append(km.inertia\_)

# Plot sse against k

plt.figure(figsize=(6, 6))

plt.plot(list\_k, sse, '-o')

plt.xlabel(r'Number of clusters \*k\*')

plt.ylabel('Sum of squared distance');