# Data Analysis and Machine Learning: Support Vector Machines

Morten Hjorth-Jensen[1,2]

Department of Physics, University of Oslo[1]

Department of Physics and Astronomy and National Superconducting Cyclotron Laboratory, Michigan State University[2]

May 30, 2018

## Support Vector Machines, overarching aims

```
import numpy as np
from sklearn.svm import SVR
import matplotlib.pyplot as plt

# Generate sample data
X = np.sort(5*np.random.rand(40,1), axis=0)
y = X**3
y=y.ravel()

# Add noise to targets
X[::4] +=3*(0.5 - np.random.rand(1))
y[::5] += 50 * (0.5 - np.random.rand(8))

plt.plot(X,y, 'g^')

#SVR Fit
svr_poly = SVR(kernel='poly', C=1e3, degree=3)
y_poly = svr_poly.fit(X, y).predict(X)

# Plots
z = np.arange(0, 5, 0.1)
t = z**3
fig = plt.figure()
ax = fig.add_subplot(111)
plt.plot(z,z**3, 'r--', label='Cubic Function with No Noise')
lw = 2
plt.scatter(X, y, color='darkorange', label='Gaussian Cubic Noise')
plt.plot(X, y_poly, color='green', lw=lw, label='Polynomial model')
```