Machine Learning

Prof. Sergei Gleyzer Lecture

PH451, PH551

January 28, 2025

Announcements

- **HS #1 due today at 1pm**
- **HS #2 this week, due on Tue. Feb. 4**
- **Textbook: Read Chapters 4, 5**

Machine Learning

Algorithm choice sets hypothesis Class H

- Goal: find the best function within H
  - eg. one that makes the fewest "mistakes"
  - Optimization problem via a learning process
- Evaluate?
  - **Loss (Risk) Function** on training data
  - Many possible loss functions:
    - Squared
    - Absolute - choice depends on the problem!
    - Cross-entropy

Regression

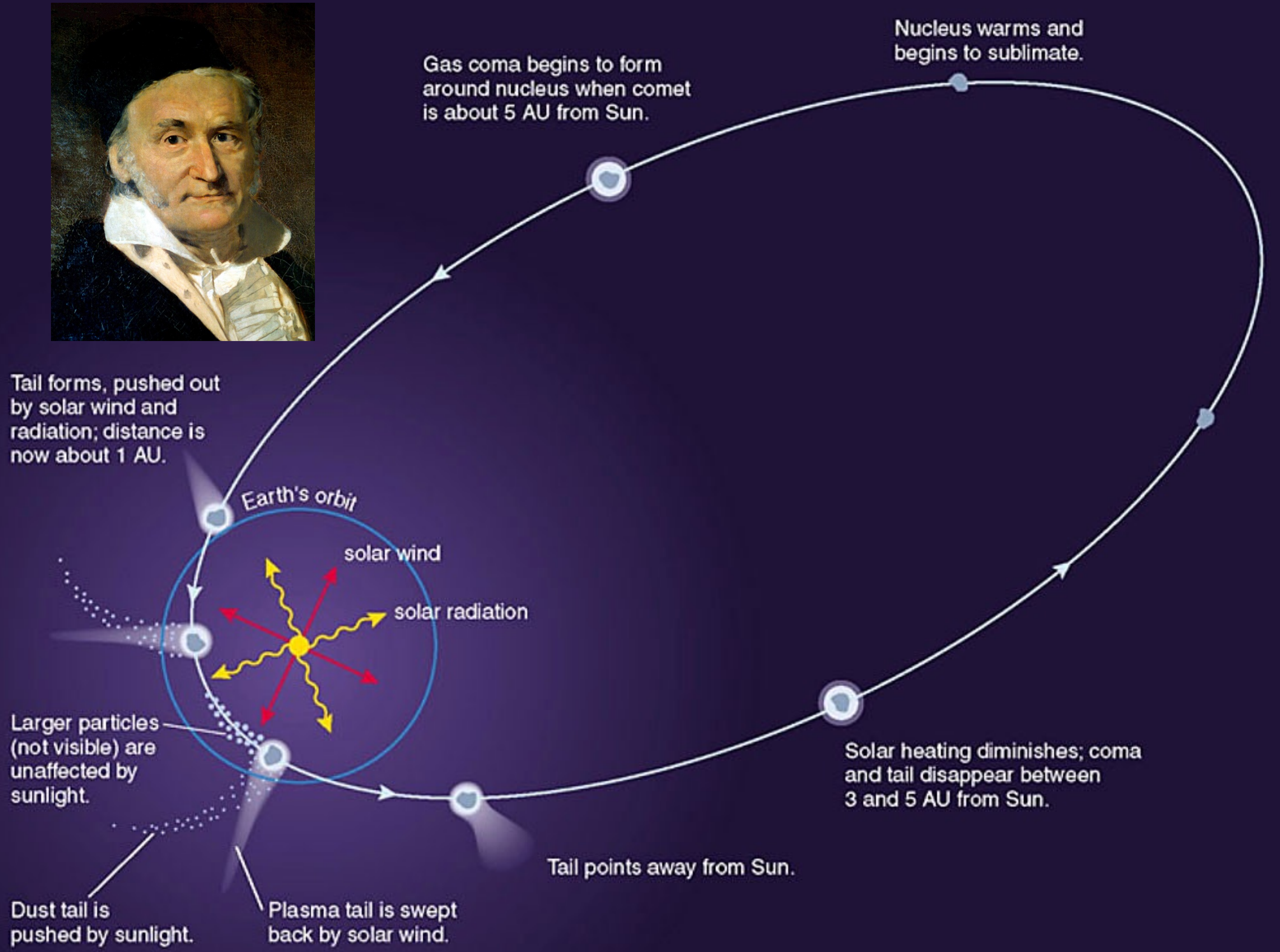
Getting Started Prediction Competition

House Prices - Advanced Regression Techniques

Predict sales prices and practice feature engineering, RFs, and gradient boosting

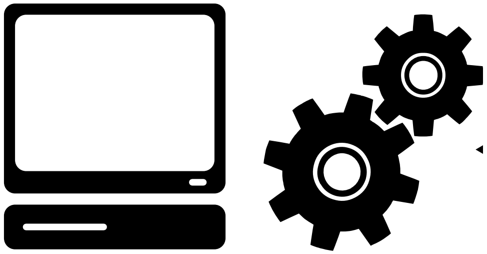
Kaggle · 4,937 teams · Ongoing

https://www.blueprintincome.com/tools/life-expectancy-calculator-how-long-will-i-live/


Nucleus warms and begins to sublimate.
Gas coma begins to form around nucleus when comet is about 5 AU from Sun.
Tail forms, pushed out by solar wind and radiation; distance is now about 1 AU.
Earth's orbit
solar wind
solar radiation
Larger particles (not visible) are unaffected by sunlight.
Dust tail is pushed by sunlight.
Plasma tail is swept back by solar wind.
Tail points away from Sun.
Solar heating diminishes; coma and tail disappear between 3 and 5 AU from Sun.

Regression

Modify evaluation in induction algorithm



Maximum separation → **Minimal variance**

Linear Regression

Linear Regression model prediction

- $y = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots + \theta_n x_n$ (Eqn. 4-1)

Vectorized form:

- $y = h_\theta(x) = \theta \cdot x$ (Eqn. 4-2)

Linear Regression

$$\mathbf{y} = \mathbf{h}_{\boldsymbol{\theta}}(\mathbf{x}) = \boldsymbol{\theta} \cdot \mathbf{x}$$

$\boldsymbol{\theta}$ is the model's parameter vector:

- bias term θ_0 and the feature weights θ_1 to θ_n

$\mathbf{x}$ is the instance's feature vector: x_0 to x_n

- x_0 always equal to 1

$\boldsymbol{\theta} \cdot \mathbf{x}$ is the **dot product** of the vectors $\boldsymbol{\theta}$ and $\mathbf{x}$

$\mathbf{h}_{\boldsymbol{\theta}}$ is the hypothesis function using the model parameters $\boldsymbol{\theta}$.

Training Linear Regression

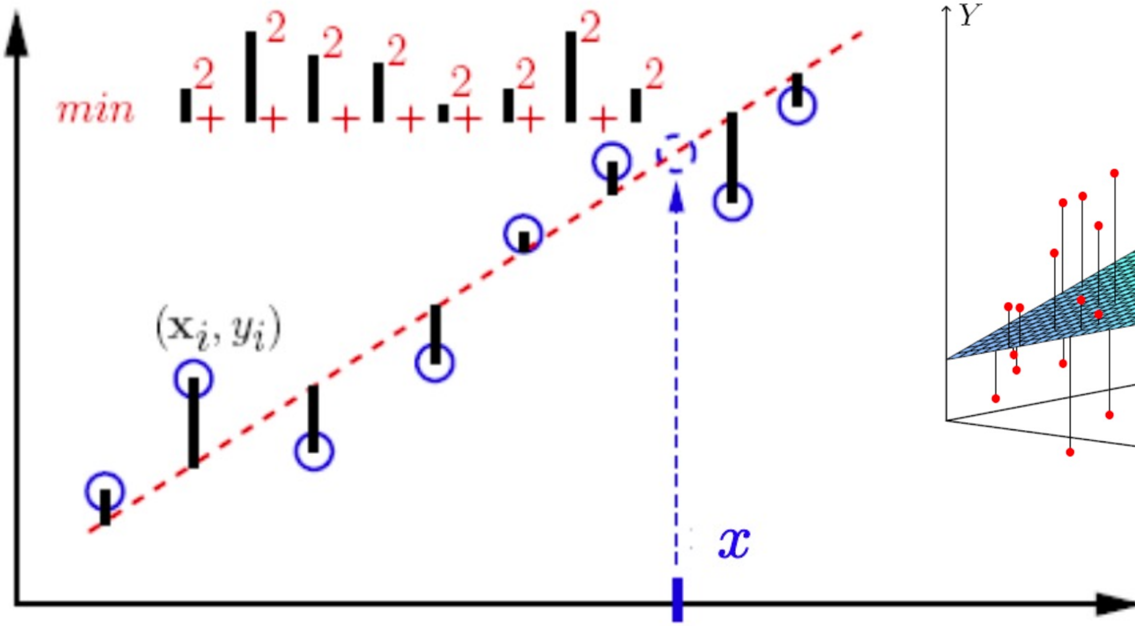
Recall that: $$\mathrm{RMSE}(\mathbf{X}, h) = \sqrt{\frac{1}{m}\sum_{i=1}^{m}\left(h\left(\mathbf{x}^{(i)}\right) - y^{(i)}\right)^2}$$

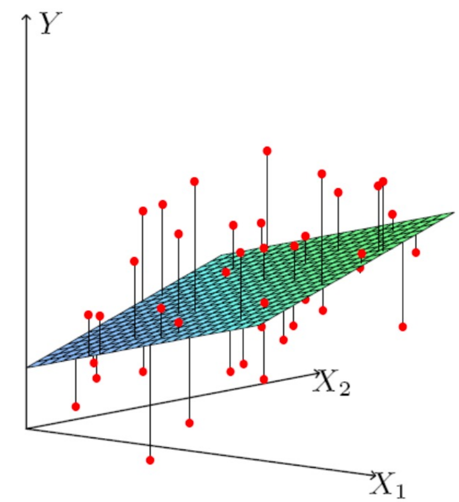
We can then **minimize**

$$\mathrm{MSE}\left(\mathbf{X}, h_{\boldsymbol{\theta}}\right) = \frac{1}{m}\sum_{i=1}^{m}\left(\boldsymbol{\theta}^{\top}\mathbf{x}^{(i)} - y^{(i)}\right)^2$$

- How well does the model **fit** the data?

Least Squares



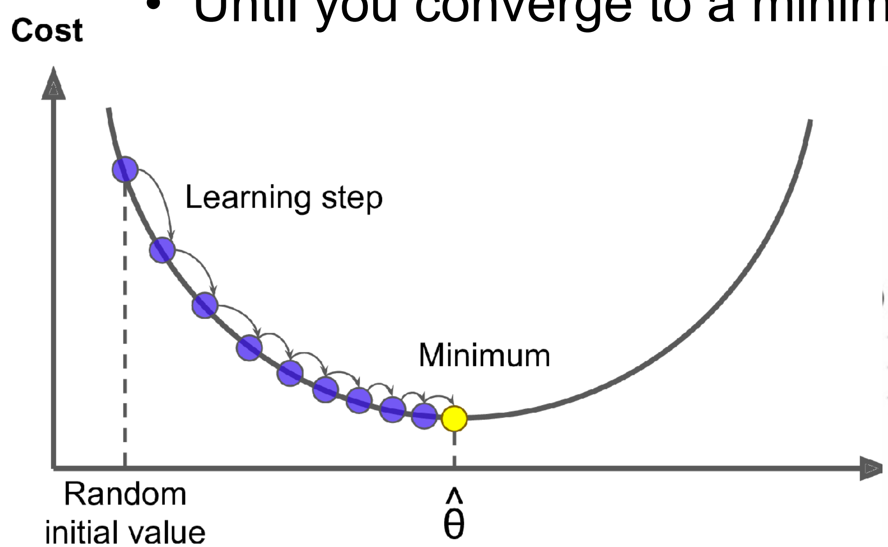


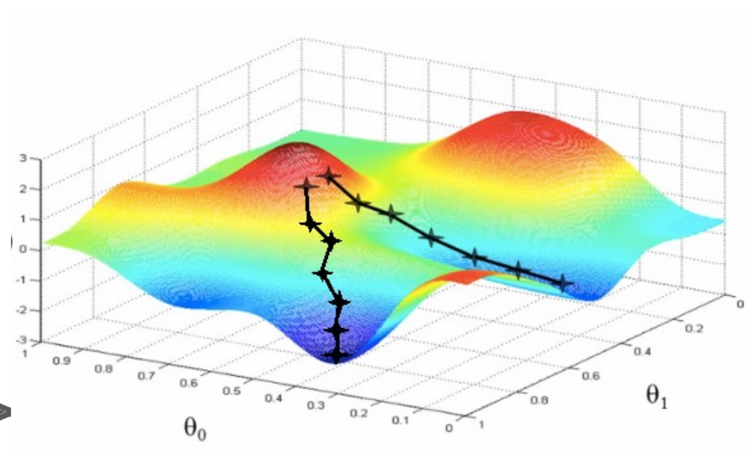
Fit by minimizing squares of errors (variance)

Gradient Descent

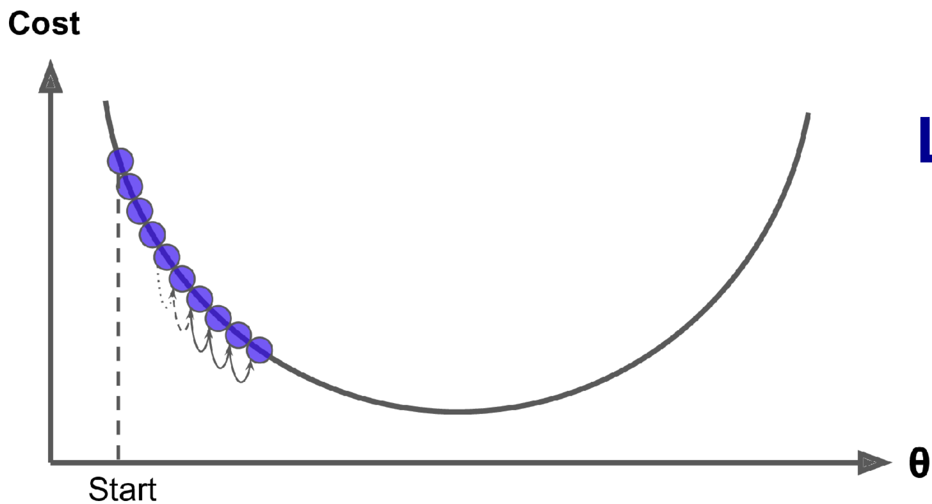
Iterative optimization algorithm to get to an optimal solution or minimize a cost function

- Measure **local gradient** of the error function
- Follow the **steepest gradient** down
  - With each step try to decrease the cost function
  - Until you converge to a minimum

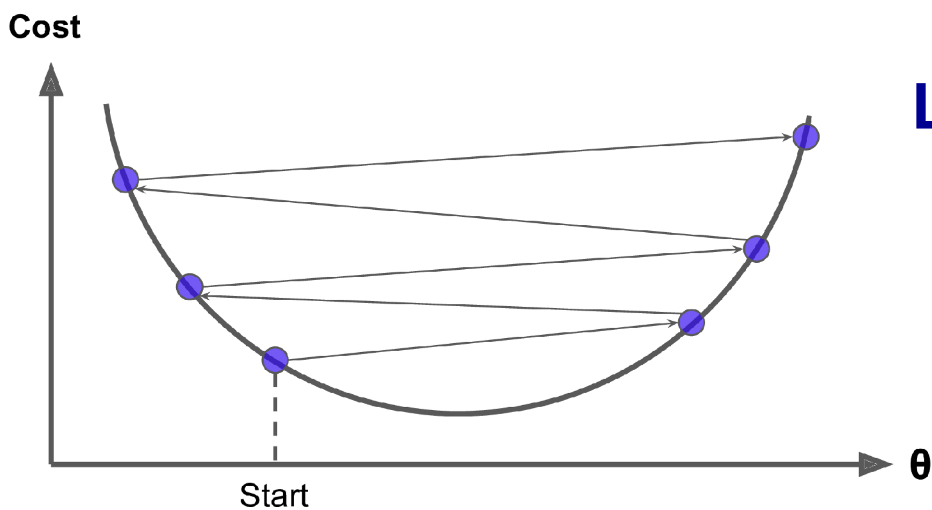




Gradient Descent

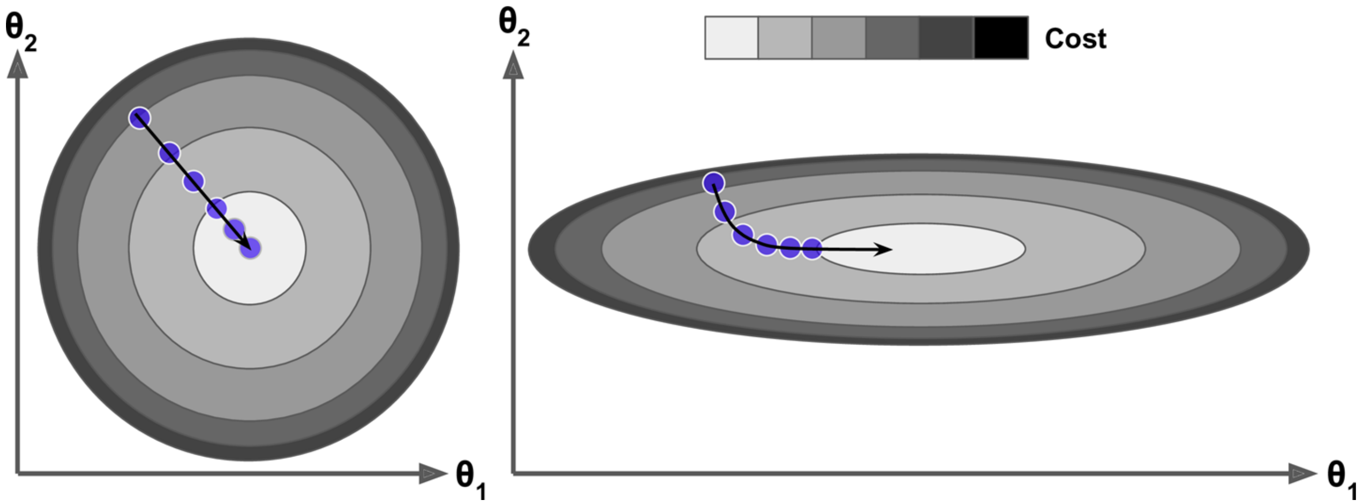


Learning rate α too small



Learning rate α too large

Feature Scaling



Improve learning by making sure all features have similar scales (feature scaling)

- **Gradient Descent converges much faster**

Feature Scaling

Example:

- Rescale features to have **zero mean** and **unit variance**, i.e.

$$x_j^{(i)} \leftarrow \frac{x_j^{(i)} - \mu_j}{s_j}$$ where **mean** $$\mu_j = \frac{1}{n}\sum_{i=1}^{n} x_j^{(i)}$$

and s_j is **standard deviation** of feature j

Gradient Descent

Batch GD

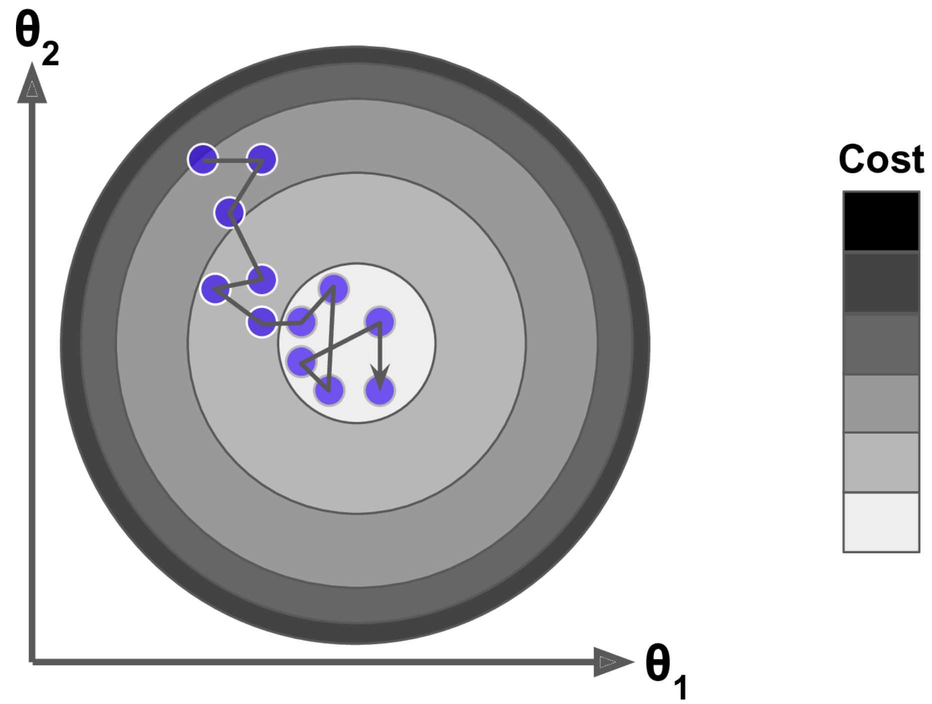
- Use full training dataset (batch)
- Can be slow

Stochastic GD

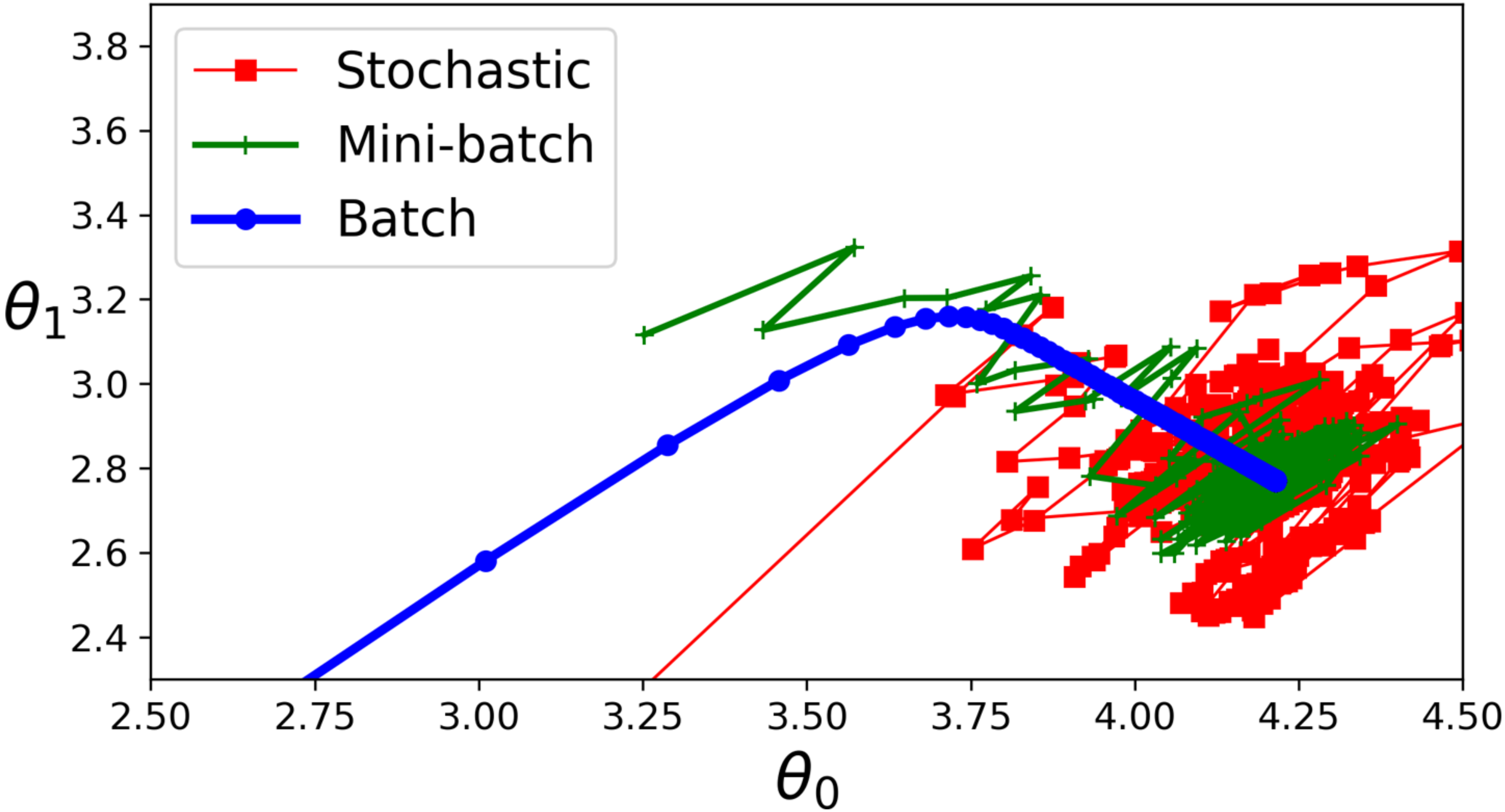
- Use random instance
  - Shuffle instances
  - Gradually reduce LR
- Fast, can get out of local minima

Mini-Batch GD

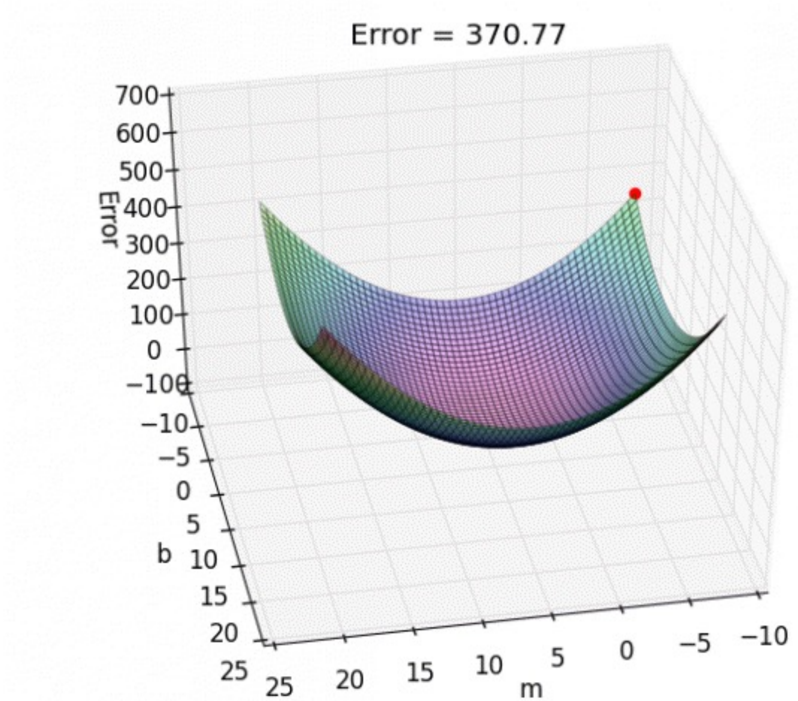
- Use mini-batches

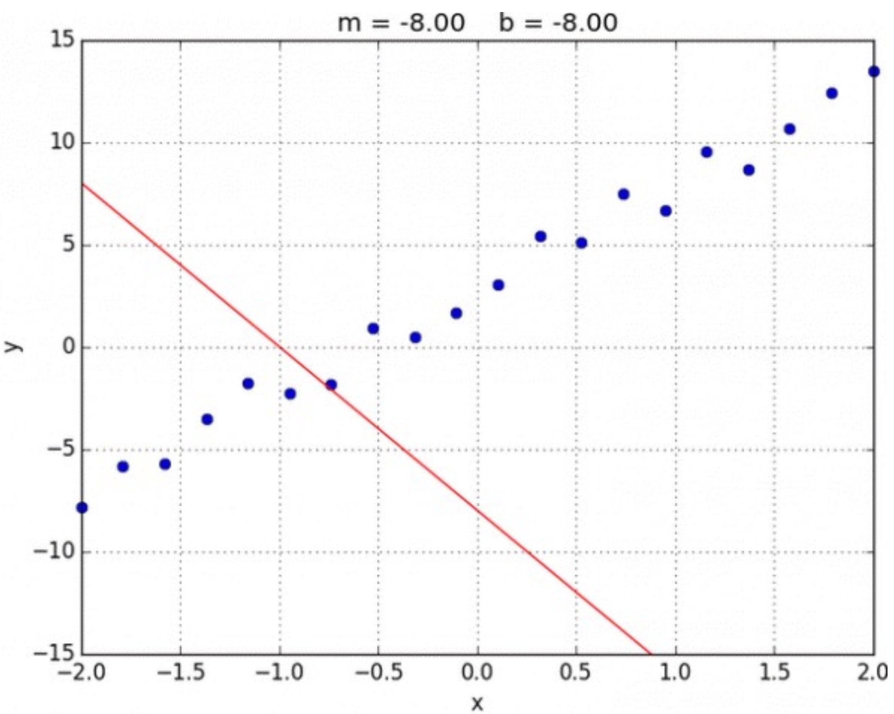


Gradient Descent

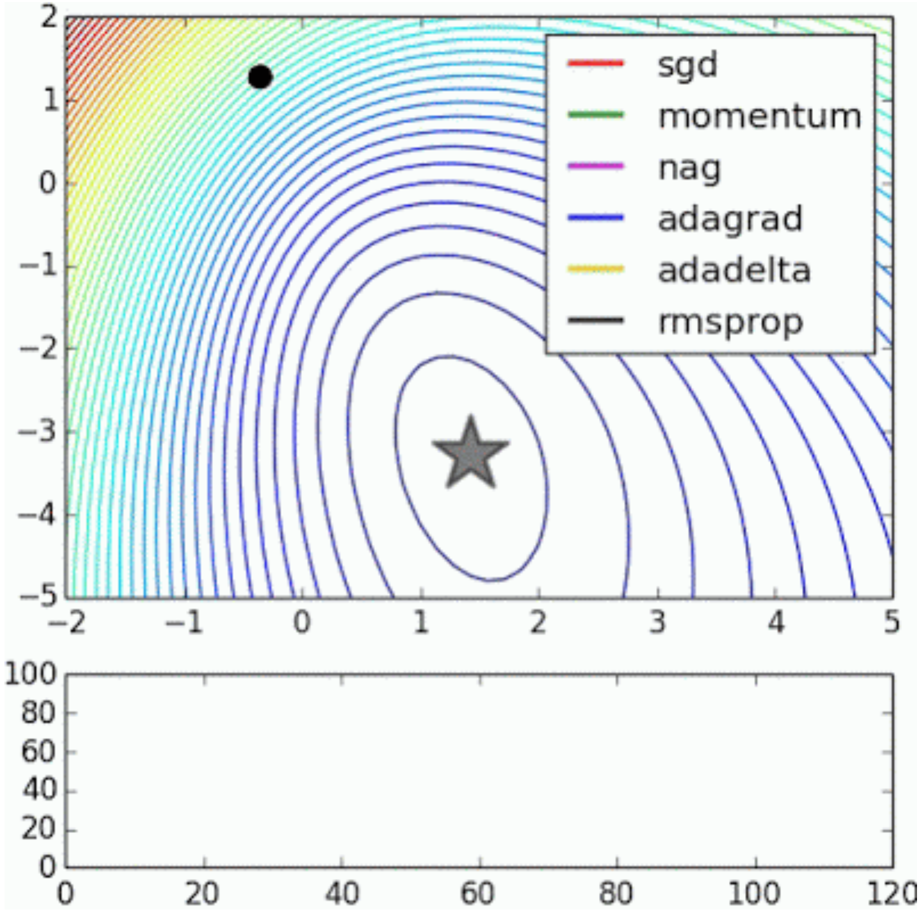


Gradient Descent



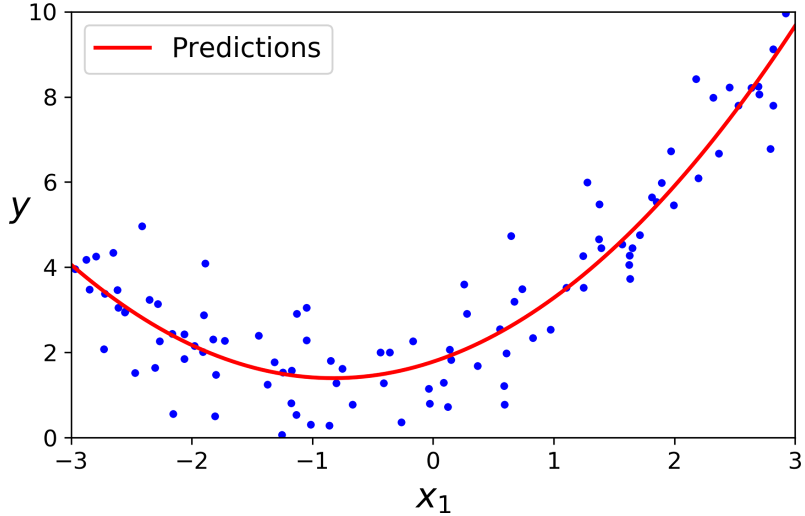


Gradient Descent



Polynomial Regression

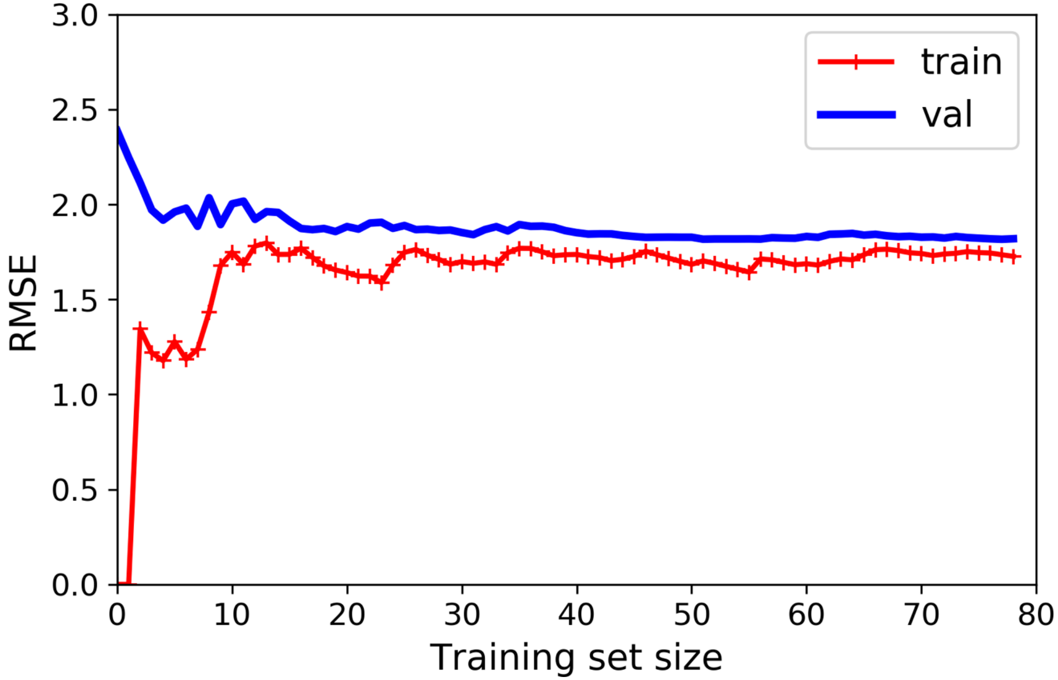
- **Train powers of features with linear regression**



- **Watch out for under or overfitting**
  - **How do you know which?**

Learning Curves

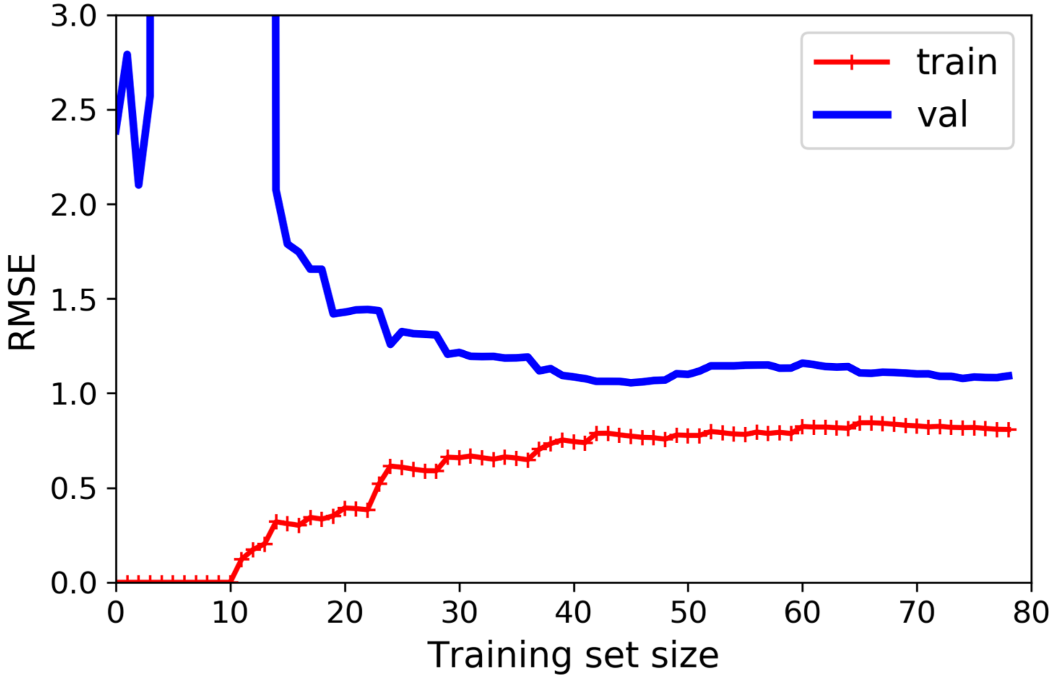
Validation vs. Training Error:



Underfitting Model

Learning Curves

Validation vs. Training Error:



Overfitting Model

Bias Variance Trade-Off

Sources of generalization error:

- **Bias:**
  - (Wrong) assumptions about data and model
- **Variance:**
  - Model DoF and sensitivity to variations in training dataset (many dof, high variance - overfitting model)
- **Resolution** (Irreducible error):
  - Noisy data

Trade-off! e.g. Increase model complexity:

- **reduce bias**, **increase variance** and vice versa

Hands-on Activity