CPSC 340:
Machine Learning and Data Mining

Ordinary Least Squares

Original version of these slides by Mark Schmidt, with modifications by Mike Gelbart.

Admin

- Assignment 1 due tonight
- Reminder: midterm in class on Wednesday Feb 14 (in 2.5 weeks)

Supervised Learning Round 2: Regression

- We're going to revisit supervised learning:

$$X = \begin{bmatrix} \quad \end{bmatrix} \qquad y = \begin{bmatrix} \quad \end{bmatrix}$$

- Previously, we considered classification:
  - We assumed y_i was discrete: y_i = 'spam' or y_i = 'not spam'.
- Now we're going to consider regression:
  - We allow y_i to be numerical: y_i = 10.34cm.

Regression examples

- We want to discover relationship between numerical variables:
  - Does number of lung cancer deaths change with number of cigarettes?
  - Does how UBC GPA relate to high school GPA?
  - Can I predict your credit score based on your age, occupation, and income?

Handling Numerical Labels

- One way to handle numerical y_i: discretize.
  - E.g., for 'age' could we use {'age ≤ 20', '20 < age ≤ 30', 'age > 30'}.
  - Now we can apply methods for classification to do regression.
  - But coarse discretization loses resolution.
  - And fine discretization requires lots of data.
- There exist regression versions of classification methods:
  - Regression trees, probabilistic models, non-parametric models.
- Today: one of oldest, but still most popular/important methods:
  - Linear regression based on squared error.
  - Very interpretable and the building block for more-complex methods.

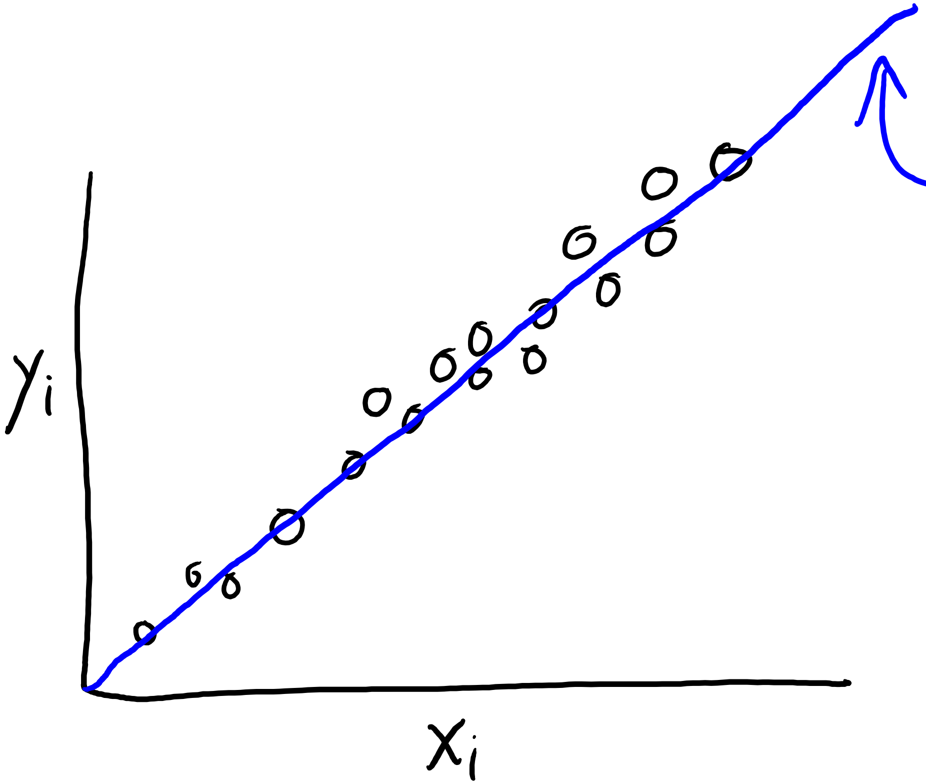
Linear Regression in 1 Dimension

- Assume we only have 1 feature (d = 1):
  – E.g., x_i is number of cigarettes and y_i is number of lung cancer deaths.
- Linear regression makes predictions $\hat{y}_i$ using a linear function of x_i:

$$\hat{y}_i = w x_i$$

- The parameter 'w' is the weight or regression coefficient of x_i.
- As x_i changes, slope 'w' affects the rate that $\hat{y}_i$ increases/decreases:
  – Positive 'w': $\hat{y}_i$ increase as x_i increases.
  – Negative 'w': $\hat{y}_i$ decreases as x_i increases.

Linear Regression in 1 Dimension



line $\hat{y}_i = wx_i$ for a particular slope 'w'.

Aside: terminology woes

- Different fields use different terminology and symbols.
  - Data points = **objects** = **examples** = rows = observations.
  - **Inputs** = predictors = **features** = explanatory variables= regressors = independent variables = covariates = columns.
  - **Outputs** = outcomes = targets = response variables = dependent variables (also called a "label" if it's categorical).
  - Regression coefficients = **weights** = parameters = betas.
- With linear regression, the symbols are inconsistent too:
  - In ML (e.g. CPSC 340), the features are X and the weights are w.
  - In statistics (e.g. STAT 306), the features are X and the weights are β.
  - In optimization (e.g. CPSC 406), the features are A and the weights are x.

Least Squares Objective

- Our linear model is given by:

$$\hat{y}_i = w x_i$$

- So we make predictions for a new example by using:

$$\hat{y}_i = w \tilde{x}_i$$

- But we can't use the same error as before:
  - Even if data comes from a linear model but has noise, we can have $\hat{y}_i \neq y_i$ for all training examples 'i' for the "best" model

Least Squares Objective

- We need a way to evaluate numerical error.
- Classic way is setting slope 'w' to minimize sum of squared errors:

$$f(w) = \sum_{i=1}^{n} (wx_i - y_i)^2$$

True value of y_i

Our prediction $\hat{y}_i$

Sum up the squared differences over all training examples.

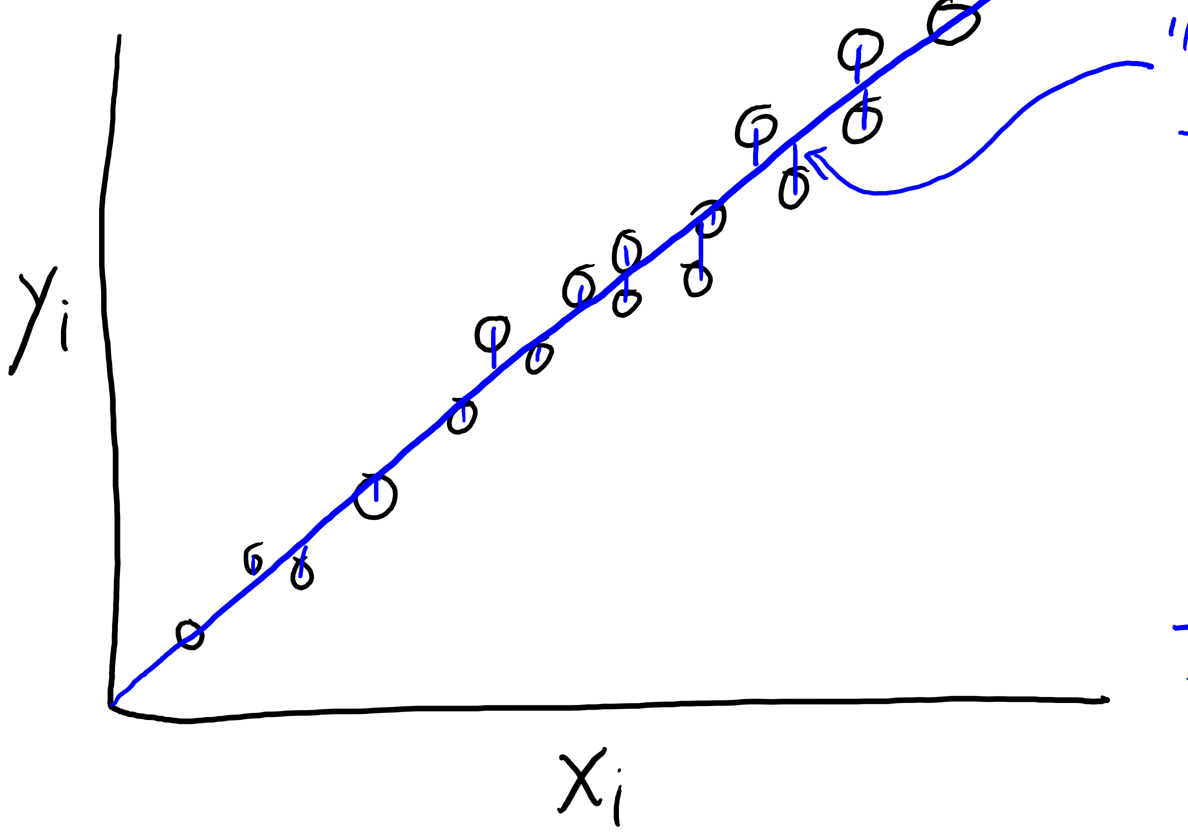
Difference between prediction and true value for example 'i'.

- There are some justifications for this choice.
  - A probabilistic interpretation is coming later in the course.
- But usually, it is done because it is easy to minimize.

Least Squares Objective

- Classic way to set slope ‘w’ is minimizing sum of squared errors:

$$f(w) = \sum_{i=1}^{n} (w x_i - y_i)^2$$



"Error" is the sum of the squared values of these vertical distances between the line (wx_i) and the targets (y_i)

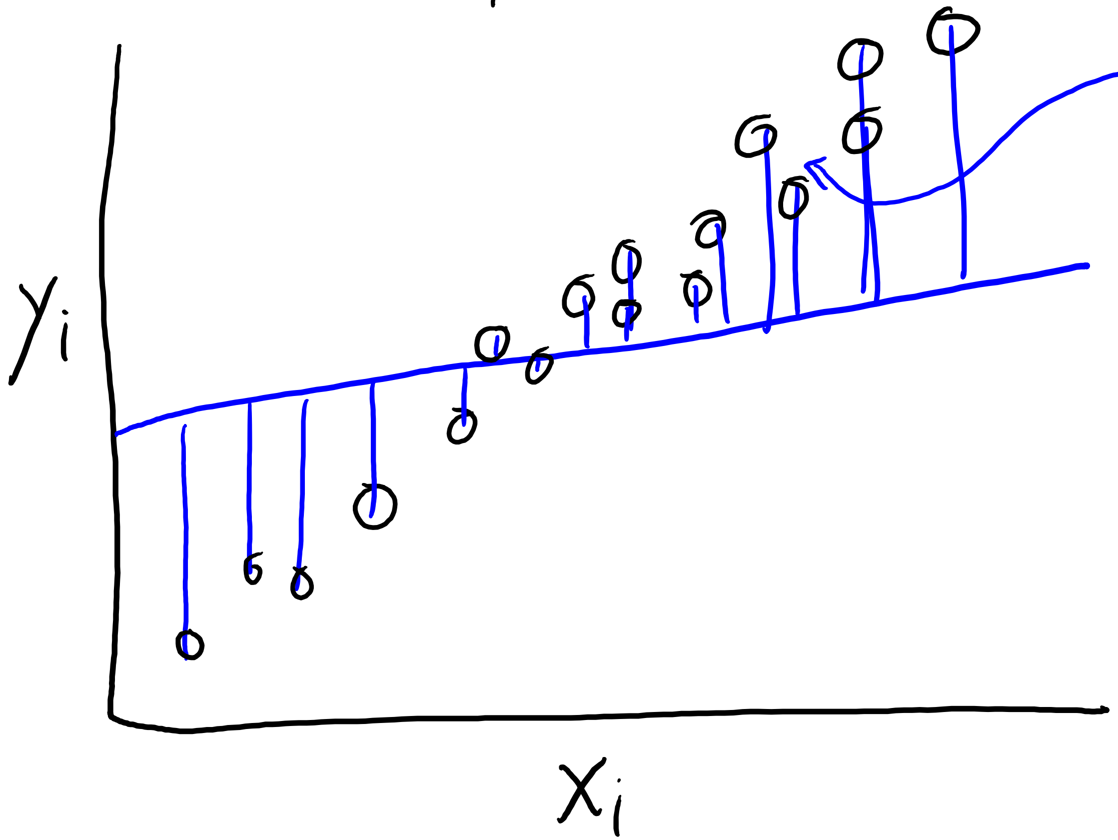
$\downarrow$

If this error is small, then our predictions are close to the targets.

Least Squares Objective

- Classic way to set slope 'w' is minimizing sum of squared errors:

$$f(w) = \sum_{i=1}^{n} (wx_i - y_i)^2$$



"Error" is the sum of the squared values of these vertical distances between the line (wx_i) and the targets (y_i)

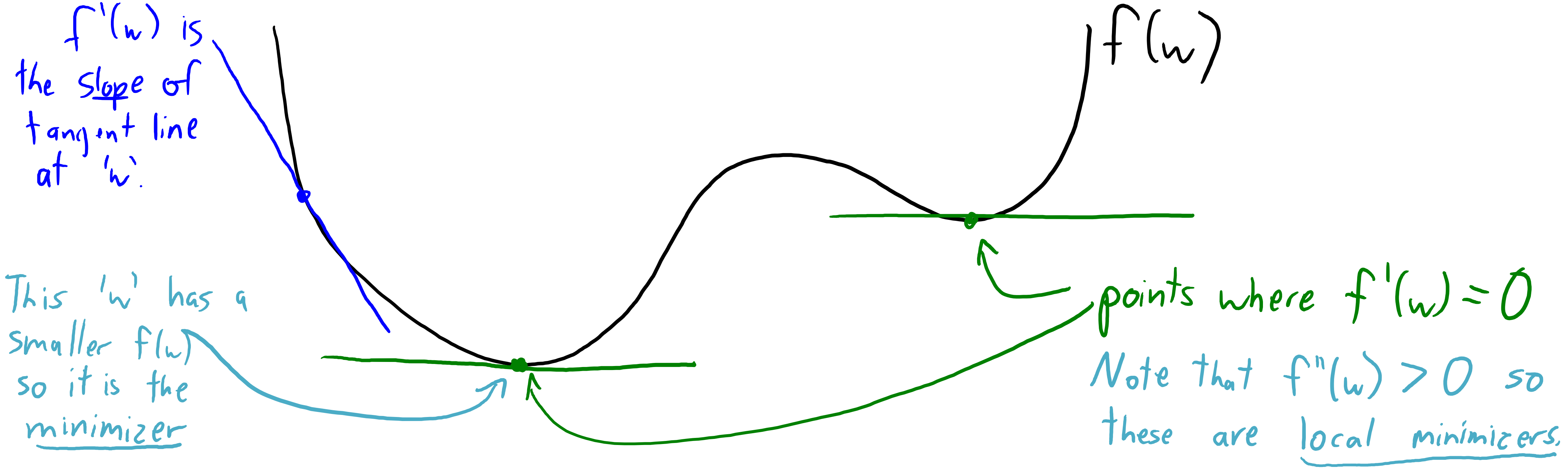
If this error is large, then our predictions are far from the targets.

Digression: Multiplying by a Positive Constant

- Note that this problem: $$f(w) = \sum_{i=1}^{n} (wx_i - y_i)^2$$
- Has the same set of minimizers as this problem:
$$f(w) = \frac{1}{2}\sum_{i=1}^{n} (wx_i - y_i)^2$$
- And these also have the same minimizers:
$$f(w) = \frac{1}{n}\sum_{i=1}^{n} (wx_i - y_i)^2 \qquad f(w) = \frac{1}{2n}\sum_{i=1}^{n} (wx_i - y_i)^2 + 1000$$
- I can multiply 'f' by any positive constant and not change solution.
  - Gradient will still be zero at the same locations.
  - We'll use this trick a lot!

Minimizing a Differential Function

- Math 101 approach to minimizing a differentiable function ‘f’:
    1. Take the derivative of ‘f’.
    2. Find points ‘w’ where the derivative f’(w) is equal to 0.
    3. Choose the smallest one (but check that f’’(w) is positive).



Finding Least Squares Solution

- Finding 'w' that minimizes sum of squared errors:

$$f(w) = \frac{1}{2}\sum_{i=1}^{n}(wx_i - y_i)^2 = \frac{1}{2}(wx_1 - y_1)^2 + \frac{1}{2}(wx_2 - y_2)^2 + \cdots + \frac{1}{2}(wx_n - y_n)^2$$

$$f'(w) = \sum_{i=1}^{n}(wx_i - y_i)x_i = (wx_1 - y_1)x_1 + (wx_2 - y_2)x_2 + \cdots + (wx_n - y_n)x_n$$

Set $f'(w) = 0$: $\sum_{i=1}^{n}(wx_i - y_i)x_i = 0$ or $\sum_{i=1}^{n}[wx_i^2 - y_i x_i] = 0$

or $\sum_{i=1}^{n} wx_i^2 = \sum_{i=1}^{n} y_i x_i$

or $w\sum_{i=1}^{n} x_i^2 = \sum_{i=1}^{n} y_i x_i$

so $\boxed{w = \dfrac{\sum_{i=1}^{n} y_i x_i}{\sum_{i=1}^{n} x_i^2}}$

Is this a minimizer?

$$f''(w) = \sum_{i=1}^{n} x_i^2$$

Since $(\text{anything})^2$ is non-negative, $f''(w) \geq 0$.

If at least one $x_i \neq 0$ then $f''(w) > 0$ and this is a minimizer.

Motivation: Combining Explanatory Variables

- Smoking is not the only contributor to lung cancer.
  - For example, environmental factors like exposure to asbestos.
- How can we model the combined effect of smoking and asbestos?
- A simple way is with a 2-dimensional linear function:

$$\hat{y}_i = w_1 x_{i1} + w_2 x_{i2}$$

Value of feature 2 in example 'i'

"weight" on feature 2.

"weight" of feature 1

Value of feature 1 in example 'i'

- We have a weight w_1 for feature '1' and w_2 for feature '2'.

Least Squares in d-Dimensions

- If we have 'd' features, the d-dimensional linear model is:

$$\hat{y}_i = w_1 x_{i1} + w_2 x_{i2} + w_3 x_{i3} + \cdots + w_d x_{id}$$

- We can re-write this in summation notation:

$$\hat{y}_i = \sum_{j=1}^{d} w_j x_{ij}$$

- We can also re-write this in vector notation:

$$\hat{y}_i = \underbrace{w^T x_i}_{\text{"inner product" between vectors}}$$

$$w^T x = \overbrace{\begin{bmatrix} w_1 & w_2 & \cdots & w_d \end{bmatrix}}^{w^T} \overbrace{\begin{bmatrix} x_{i1} \\ x_{i2} \\ \vdots \\ x_{id} \end{bmatrix}}^{x_i} = \sum_{j=1}^{d} w_j x_{ij}$$

- In words, our model is that the output is a weighted sum of the inputs.

Notation Alert (again)

- In this course, all vectors are assumed to be column-vectors:

$$w = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_d \end{bmatrix} \qquad y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \qquad x_i = \begin{bmatrix} x_{i1} \\ x_{i2} \\ \vdots \\ x_{id} \end{bmatrix}$$

- So w^Tx_i is a scalar:

$$w^T x_i = \begin{bmatrix} w_1 & w_2 & \cdots & w_d \end{bmatrix} \begin{bmatrix} x_{i1} \\ x_{i2} \\ \vdots \\ x_{id} \end{bmatrix} = w_1 x_{i1} + w_2 x_{i2} + \cdots + w_d x_{id}$$

$$= \sum_{j=1}^{d} w_j x_{id}$$

- So rows of 'X' are actually transpose of column-vector x_i:

$$X = \begin{bmatrix} — & x_1^T & — \\ — & x_2^T & — \\ & \vdots & \\ — & x_n^T & — \end{bmatrix}$$

Least Squares in d-Dimensions

- The linear least squares model in d-dimensions minimizes:

$$f(w) = \frac{1}{2}\sum_{i=1}^{n}(\underbrace{w^Tx_i}_{\text{prediction is inner product of 'w' and 'x}_i\text{' (linear combination of features)}} - y_i)^2$$

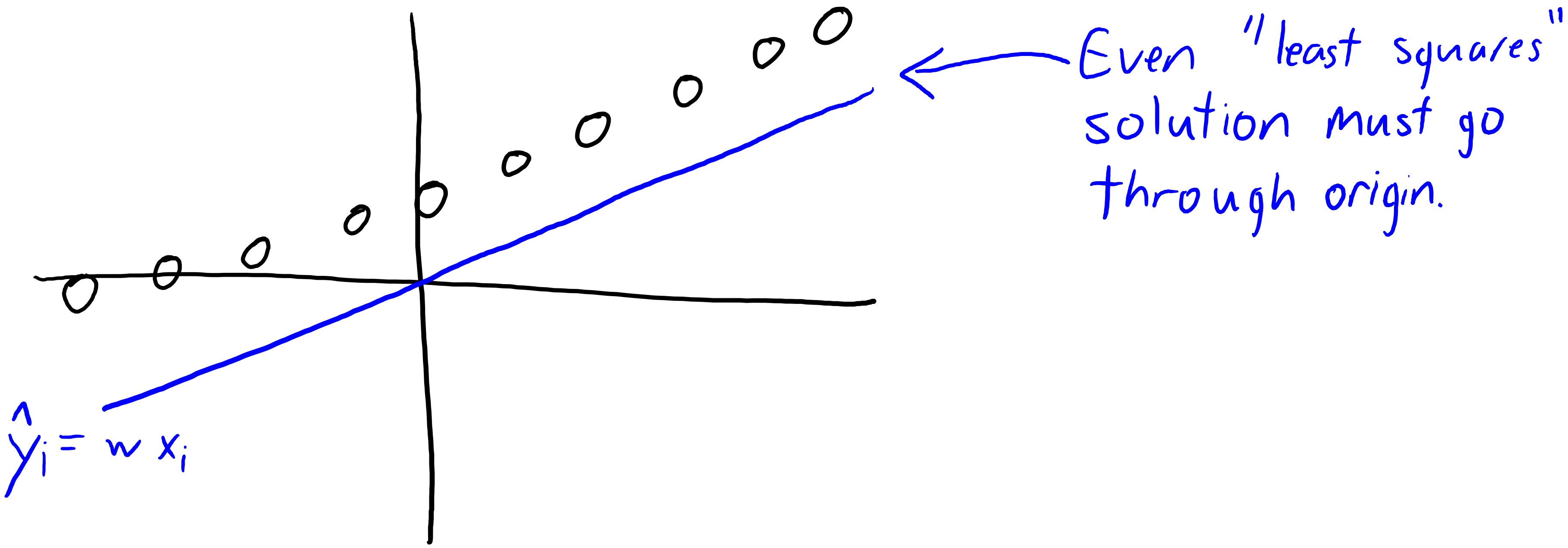
'w' is now a vector

"Error" is still the sum of squared differences between "true" y_i and our "prediction" w^Tx_i

- How do we find the best vector 'w'?
  - Set the derivative of each variable ("partial derivative") to 0?
  - We'll go through this next class.
  - But first…

Modeling a y-intercept?

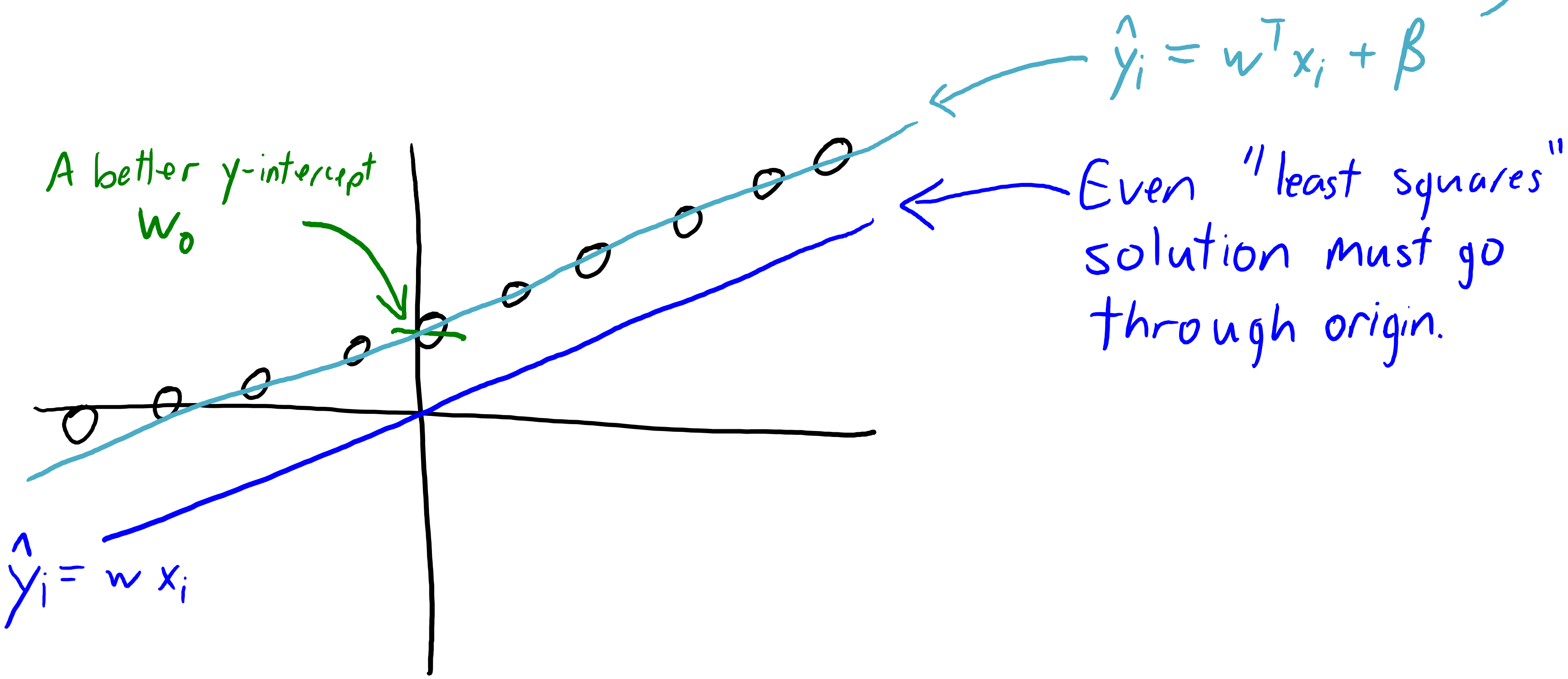
- Linear model is $\hat{y}_i = wx_i$ instead of $\hat{y}_i = wx_i + \beta$ with y-intercept β.
- Without an intercept, if $x_i = 0$ then we must predict $\hat{y}_i = 0$.



Modeling a y-intercept?

- Linear model is $\hat{y}_i = wx_i$ instead of $\hat{y}_i = wx_i + \beta$ with y-intercept β.
- Without an intercept, if $x_i = 0$ then we must predict $\hat{y}_i = 0$.

Adding y-intercept fixes this.



Adding a Bias Variable

- Simple trick to add a y-intercept ("bias") variable:
  - Make a new matrix "Z" with an extra feature that is always "1".

$$X = \begin{bmatrix} 0.1 & -0.3 \\ 0.5 & 0.2 \\ 0.2 & 0.3 \end{bmatrix} \qquad Z = \begin{bmatrix} 1 & \underbrace{\begin{matrix} 0.1 & -0.2 \\ 0.5 & 0.2 \\ 0.2 & 0.3 \end{matrix}}_{X} \end{bmatrix}$$

- Now use "Z" as features in linear regression.
  - Gives a model with weights 'v' that have a non-zero y-intercept:

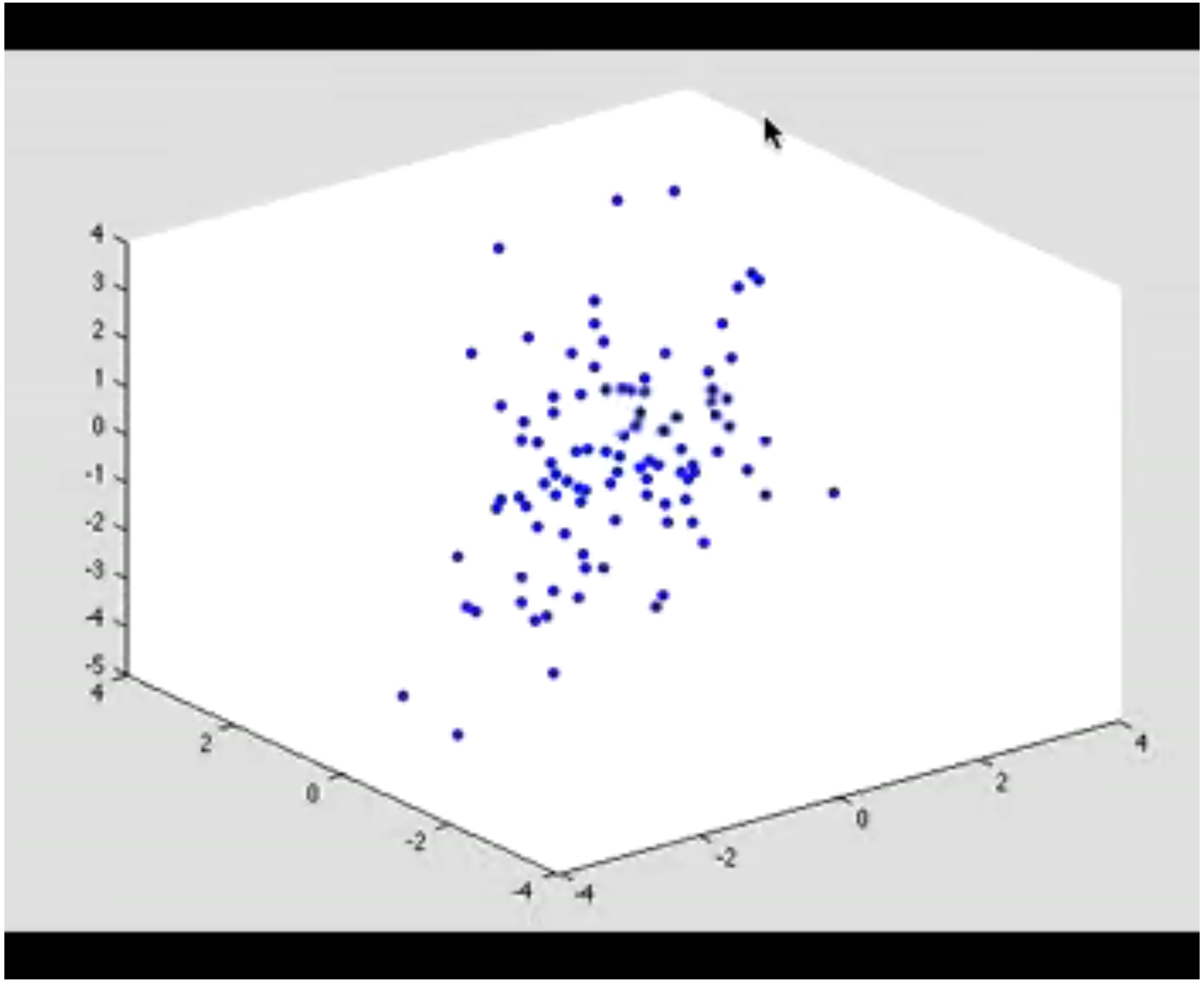
$$\hat{y}_i = \underbrace{v_1}_{\text{"}\beta\text{"}} \underbrace{z_{i1}}_{\text{"}1\text{"}} + \underbrace{v_2}_{\text{"}w_1\text{"}} \underbrace{z_{i2}}_{x_{i1}} + \underbrace{v_3}_{\text{"}w_2\text{"}} \underbrace{z_{i3}}_{x_{i2}} = \underbrace{\beta + w_1 x_{i1} + w_2 x_{i2}}_{\text{linear regression with y-intercept } \beta.}$$

- So we can have a non-zero y-intercept by changing features.
  - This means we can ignore the y-intercept in our derivations, which is cleaner.

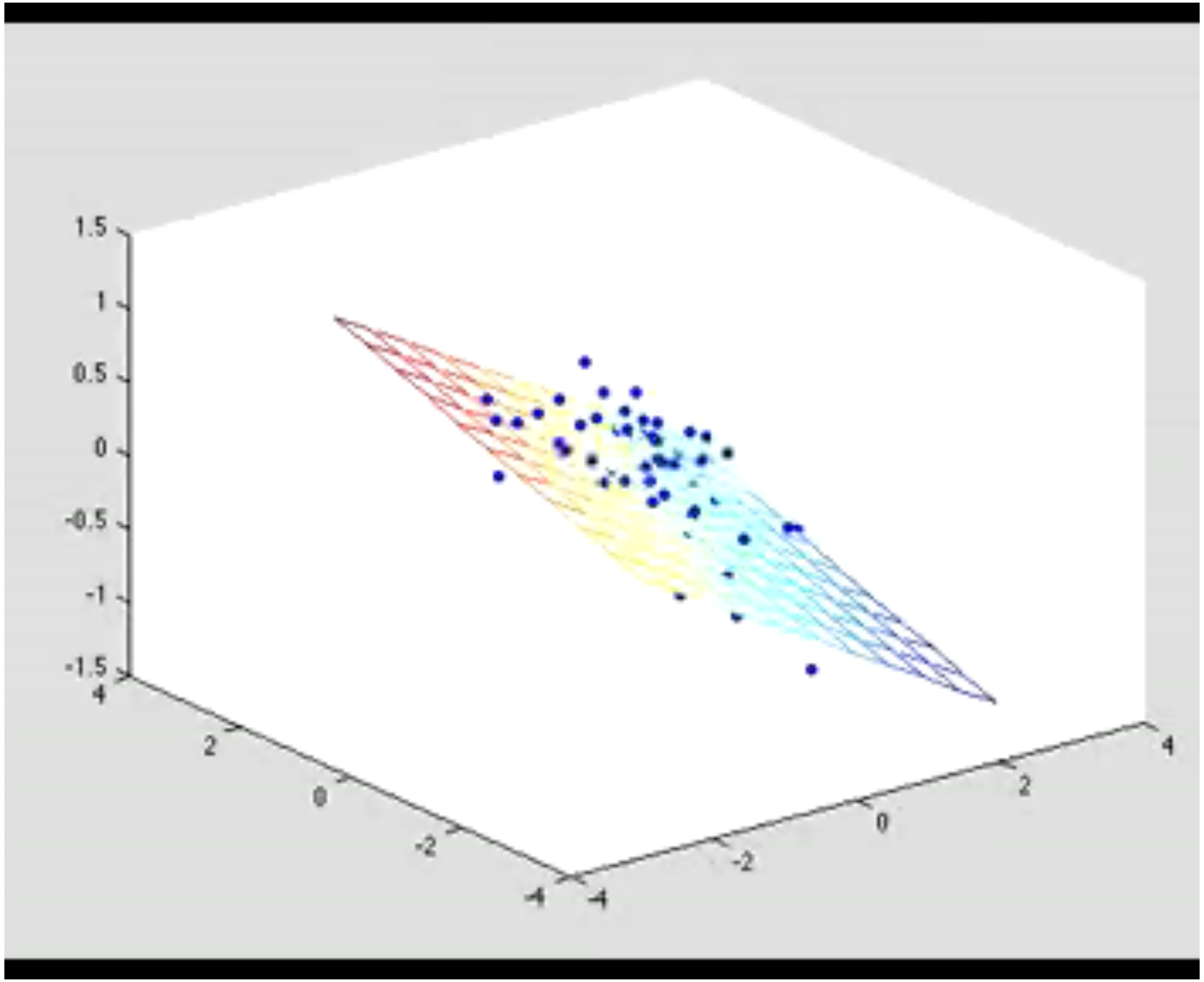
Summary

- Regression considers the case of a numerical y_i.
- Least squares is a classic method for fitting linear models.
  - With 1 feature, it has a simple closed-form solution.
- Gradient is vector containing partial derivatives of all variables.

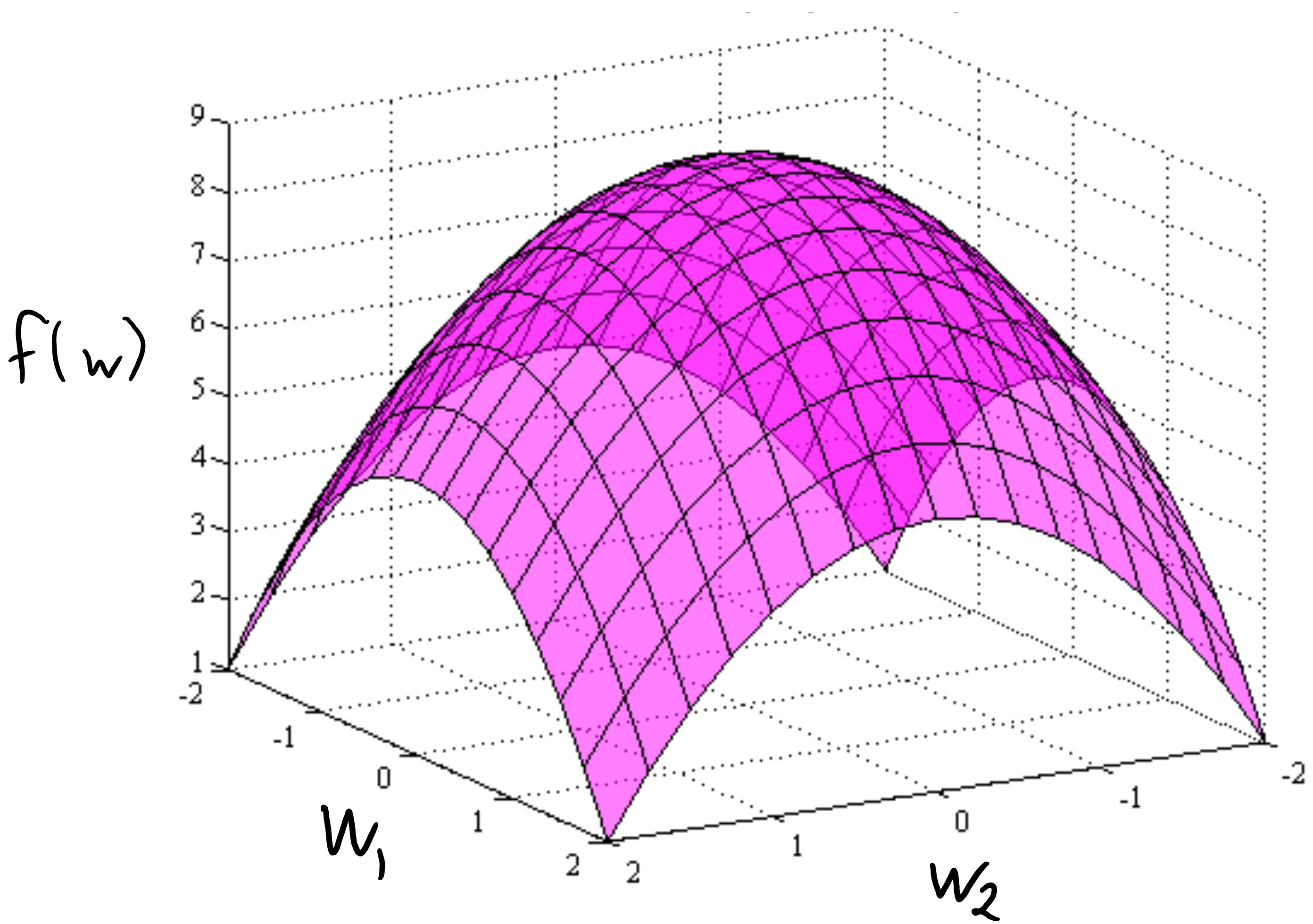
Least Squares in 2-Dimensions



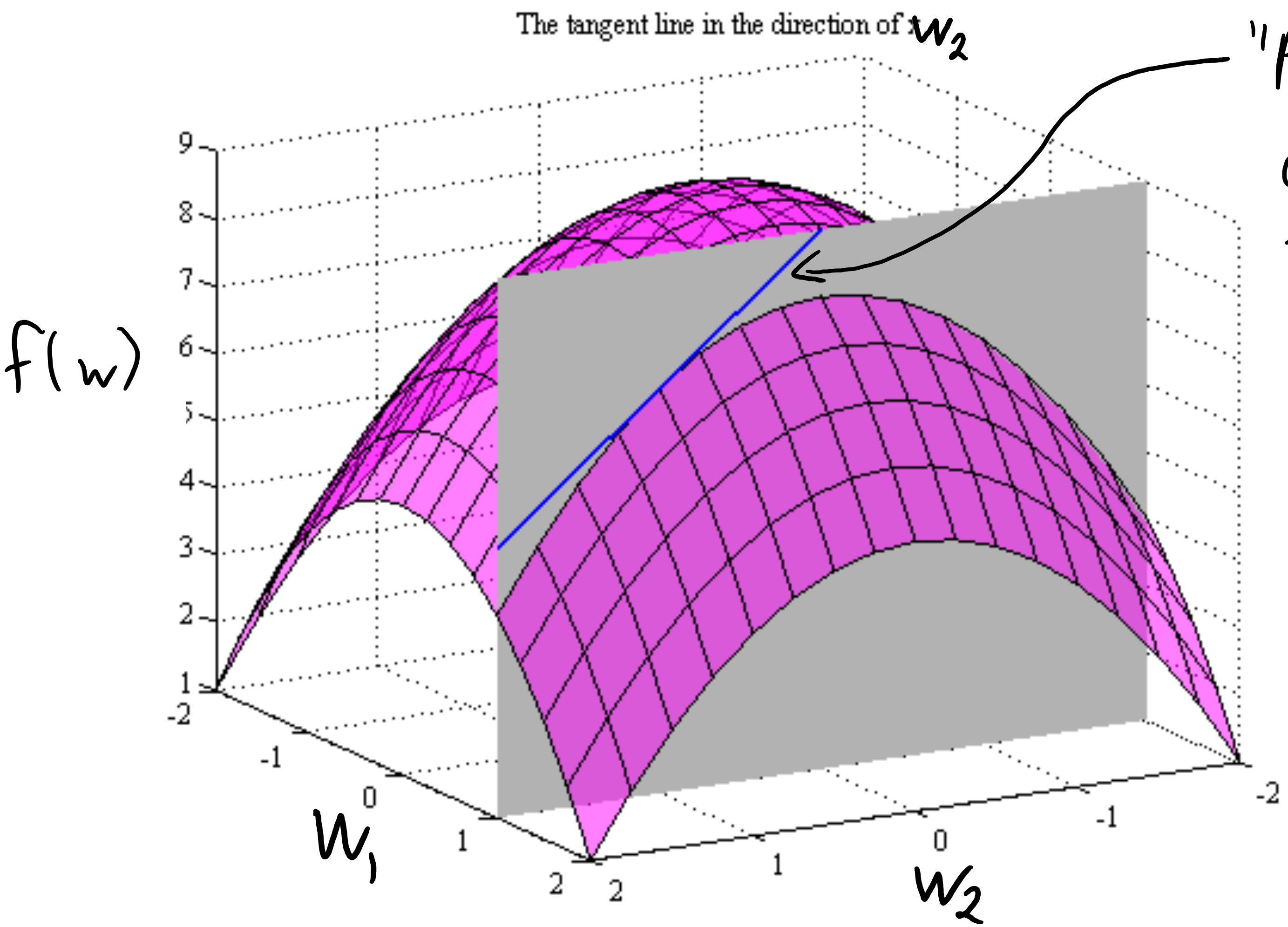
Least Squares in 2-Dimensions



Partial Derivatives



Partial Derivatives



"Partial" derivative of 'f' with respect to w_2 is the derivative with respect to w when all other variables are held fixed.

Denoted by $\frac{\partial}{\partial w_2}$ for variable w_2