

# Regression + Assignments 1, 2

(Neural Networks Implementation and Application Tutorial)

Vilém Zouhar, Noon Pokaratsiri Goldstein

17th November 2021

# Overview

- Assignment 1
- Regression
- Assignment 2

# Update

Up to 2 points for participation.

# Assignment 1

## Organization

- Late submissions ( $>10$ mins) will not be accepted unless previously agreed upon
- Other questions?

## Notes

- Very nice solutions!
- Reconstruction error on original space, not standardized (we did the same mistake 😬)

# Assignment 1

- *Tutor cue*: go through the assignment
- Questions?
- Did it work?
- Were you able to collaborate?

# Regression

- What is the difference between classification and regression? 🤔

# Regression

- What is the difference between classification and regression? 🤔
- What is regression in terms of functions? 🤔

# Regression

- What is the difference between classification and regression? 🤔
- What is regression in terms of functions? 🤔
  - ▶ Any function  $f : F \rightarrow \mathbb{R}$  (from joint feature space to numbers)



# Regression

- What is the difference between classification and regression? 🤔
- What is regression in terms of functions? 🤔
  - ▶ Any function  $f : F \rightarrow \mathbb{R}$  (from joint feature space to numbers)
- What is *linear* regression? 🤔

# Regression

- What is the difference between classification and regression? 🤔
- What is regression in terms of functions? 🤔
  - ▶ Any function  $f : F \rightarrow \mathbb{R}$  (from joint feature space to numbers)
- What is *linear* regression? 🤔
  - ▶  $\hat{y} = x^T \cdot \beta + \beta_c$  (parameters  $\beta, \beta_c$ )

# Regression

- What is the difference between classification and regression? 🤔
- What is regression in terms of functions? 🤔
  - ▶ Any function  $f : F \rightarrow \mathbb{R}$  (from joint feature space to numbers)
- What is *linear* regression? 🤔
  - ▶  $\hat{y} = x^T \cdot \beta + \beta_c$  (parameters  $\beta, \beta_c$ )

# Regression

- What is the difference between classification and regression? 🤔
- What is regression in terms of functions? 🤔
  - ▶ Any function  $f : F \rightarrow \mathbb{R}$  (from joint feature space to numbers)
- What is *linear* regression? 🤔
  - ▶  $\hat{y} = x^T \cdot \beta + \beta_c$  (parameters  $\beta, \beta_c$ )

Which of the following are regression (and linear/polynomial) models? 🤔 1. 5

- 2  $4 \cdot x_1 + 5$
- 3  $4 \cdot x_1 + 3 \cdot x_2^2 + 5$
- 4  $4 \cdot x_1 + 3 \cdot x_1 \cdot x_2 + 5$
- 5  $4 \cdot x_1 + 3 \cdot \sin(x_2^2) + 5$
- 6  $\begin{cases} 4 \cdot x_1 + 5 & \text{if } x_2 \geq 10 \\ 3 \cdot x_1 + 4 & \text{if } x_2 < 10 \end{cases}$

# Regression

## Regression to Classification 🤔 🤔

Assume that we have a function that outputs a score for every class, e.g. *Predict sentiment into (positive, negative, neutral)*:

$(15.0, -2.3, 4.1)$

- How do we use this for classification?

# Regression

## Regression to Classification 🤔 🤔

Assume that we have a function that outputs a score for every class, e.g. *Predict sentiment into (positive, negative, neutral)*:

$(15.0, -2.3, 4.1)$

- How do we use this for classification?
  - ▶ Argmax

# Regression

## Regression to Classification 🤔 🤔

Assume that we have a function that outputs a score for every class, e.g. *Predict sentiment into (positive, negative, neutral)*:

$(15.0, -2.3, 4.1)$

- How do we use this for classification?
  - ▶ Argmax
- Can we get a probability distribution?

# Regression

## Regression to Classification 🤔 🤔

Assume that we have a function that outputs a score for every class, e.g. *Predict sentiment into (positive, negative, neutral)*:

(15.0, -2.3, 4.1)

- How do we use this for classification?
  - ▶ Argmax
- Can we get a probability distribution?
  - ▶ Softmax:  $\frac{\exp x_i}{\sum_k \exp x_k}$



# Loss & Regularization

## Loss

- Why  $L_2$  and not  $L_1$ ?

## Regularization

# Loss & Regularization

## Loss

- Why  $L_2$  and not  $L_1$ ?
  - ▶ We care about points that are drastically mispredicted, e.g.  $L_2(-1, 10)$  and not about almost correctly predicted instances  $L_2(-1, -1.1)$

## Regularization

# Loss & Regularization

## Loss

- Why  $L_2$  and not  $L_1$ ?
  - ▶ We care about points that are drastically mispredicted, e.g.  $L_2(-1, 10)$  and not about almost correctly predicted instances  $L_2(-1, -1.1)$
  - ▶ In  $L_1$ , these would have the same weight (gradient)

## Regularization

# Loss & Regularization

## Loss

- Why  $L_2$  and not  $L_1$ ?
  - ▶ We care about points that are drastically mispredicted, e.g.  $L_2(-1, 10)$  and not about almost correctly predicted instances  $L_2(-1, -1.1)$
  - ▶ In  $L_1$ , these would have the same weight (gradient)
  - ▶  $L_1$  leads to some parameters being 0 (sometimes a good thing)

## Regularization

# Loss & Regularization

## Loss

- Why  $L_2$  and not  $L_1$ ?
  - ▶ We care about points that are drastically mispredicted, e.g.  $L_2(-1, 10)$  and not about almost correctly predicted instances  $L_2(-1, -1.1)$
  - ▶ In  $L_1$ , these would have the same weight (gradient)
  - ▶  $L_1$  leads to some parameters being 0 (sometimes a good thing)

## Regularization

- Why do we want to regularize?

# Loss & Regularization

## Loss

- Why  $L_2$  and not  $L_1$ ?
  - ▶ We care about points that are drastically mispredicted, e.g.  $L_2(-1, 10)$  and not about almost correctly predicted instances  $L_2(-1, -1.1)$
  - ▶ In  $L_1$ , these would have the same weight (gradient)
  - ▶  $L_1$  leads to some parameters being 0 (sometimes a good thing)

## Regularization

- Why do we want to regularize?
  - ▶ Prevent overfitting, prevent reliance on noise

# Loss & Regularization

## Loss

- Why  $L_2$  and not  $L_1$ ?
  - ▶ We care about points that are drastically mispredicted, e.g.  $L_2(-1, 10)$  and not about almost correctly predicted instances  $L_2(-1, -1.1)$
  - ▶ In  $L_1$ , these would have the same weight (gradient)
  - ▶  $L_1$  leads to some parameters being 0 (sometimes a good thing)

## Regularization

- Why do we want to regularize?
  - ▶ Prevent overfitting, prevent reliance on noise
- Ridge regression uses  $L_2$  penalty: *minimize*  $\arg \min L_2^2(\hat{Y}, Y) + \lambda \|\beta\|_2^2$

# Loss & Regularization

## Loss

- Why  $L_2$  and not  $L_1$ ?
  - ▶ We care about points that are drastically mispredicted, e.g.  $L_2(-1, 10)$  and not about almost correctly predicted instances  $L_2(-1, -1.1)$
  - ▶ In  $L_1$ , these would have the same weight (gradient)
  - ▶  $L_1$  leads to some parameters being 0 (sometimes a good thing)

## Regularization

- Why do we want to regularize?
  - ▶ Prevent overfitting, prevent reliance on noise
- Ridge regression uses  $L_2$  penalty: *minimize*  $\arg \min L_2^2(\hat{Y}, Y) + \lambda \|\beta\|_2^2$
- Lasso regression uses  $L_1$  penalty: *minimize*  $\arg \min L_2^2(\hat{Y}, Y) + \lambda \|\beta\|_1$



# Loss & Regularization

## Loss

- Why  $L_2$  and not  $L_1$ ?
  - ▶ We care about points that are drastically mispredicted, e.g.  $L_2(-1, 10)$  and not about almost correctly predicted instances  $L_2(-1, -1.1)$
  - ▶ In  $L_1$ , these would have the same weight (gradient)
  - ▶  $L_1$  leads to some parameters being 0 (sometimes a good thing)

## Regularization

- Why do we want to regularize?
  - ▶ Prevent overfitting, prevent reliance on noise
- Ridge regression uses  $L_2$  penalty: *minimize*  $\arg \min L_2^2(\hat{Y}, Y) + \lambda \|\beta\|_2^2$
- Lasso regression uses  $L_1$  penalty: *minimize*  $\arg \min L_2^2(\hat{Y}, Y) + \lambda \|\beta\|_1$
- ElasticNet regression uses both: *minimize*  $\arg \min L_2^2(\hat{Y}, Y) + \lambda_1 \|\beta\|_1 + \lambda_2 \|\beta\|_2^2$

# Model Capacity

- What contributes to overfitting?

# Model Capacity

- What contributes to overfitting?
  - ▶ Overfitting: little data, large model capacity, too many optimization steps

# Model Capacity

- What contributes to overfitting?
  - ▶ Overfitting: little data, large model capacity, too many optimization steps
  - ▶ Underfitting: not enough optimization steps, too strict regularization

# Model Capacity

- What contributes to overfitting?
  - ▶ Overfitting: little data, large model capacity, too many optimization steps
  - ▶ Underfitting: not enough optimization steps, too strict regularization

# Model Capacity

- What contributes to overfitting?
  - ▶ Overfitting: little data, large model capacity, too many optimization steps
  - ▶ Underfitting: not enough optimization steps, too strict regularization

## Bias-variance tradeoff

- Large bias corresponds to ...?

# Model Capacity

- What contributes to overfitting?
  - ▶ Overfitting: little data, large model capacity, too many optimization steps
  - ▶ Underfitting: not enough optimization steps, too strict regularization

## Bias-variance tradeoff

- Large bias corresponds to ... ?
  - ▶ Underfitting/small model capacity

# Model Capacity

- What contributes to overfitting?
  - ▶ Overfitting: little data, large model capacity, too many optimization steps
  - ▶ Underfitting: not enough optimization steps, too strict regularization

## Bias-variance tradeoff

- Large bias corresponds to ...?
  - ▶ Underfitting/small model capacity
- Large variance corresponds to ...?



# Model Capacity

- What contributes to overfitting?
  - ▶ Overfitting: little data, large model capacity, too many optimization steps
  - ▶ Underfitting: not enough optimization steps, too strict regularization

## Bias-variance tradeoff

- Large bias corresponds to ...?
  - ▶ Underfitting/small model capacity
- Large variance corresponds to ...?
  - ▶ Overfitting

# Assignment 2

- Any questions?