



دانشکده مهندسی برق و کامپیوتر

استاد درس:

**دستورکار آزمایشگاه هوش محاسباتی**  
جلسه ۳ کمترین مربعات، رگرسیون ریدج، بیش برآزش

## اهداف این جلسه

شما در این جلسه یاد خواهید گرفت که :

- چگونه تابع کمترین مربعات<sup>۱</sup> را پیاده سازی و خطایابی کنید
- گونه توابع پایه را پیاده سازی، خطایابی و مصورسازی نمایید.
- مفهوم بیش برازش<sup>۲</sup> را متوجه شوید.
- چگونه رگرسیون ریدج<sup>۳</sup> را پیاده سازی نمایید.

در این جلسه از پایگاه داده `height-weight-genders.csv` در کنار پایگاه داده جدید `dataEx3.csv` استفاده خواهیم کرد. همچنین کدهای نمونه و کمکی برای شما آماده شده است. در طول این جلسه، شما بر روی فایل `ex03.ipynb` کار خواهید کرد. وظیفه شما پیاده سازی توابع مورد نیاز در این فایل است.

## ۱ کمترین مربعات و مدل های توابع پایه خطی

۴

## ۱.۱ Least Squares

## تمرین اول

- تابع `least_squares(y,tx)` که راه حل معادلات نرمالی که در کلاس یادگرفته اید را پیاده سازی کرده است، تکمیل کنید. این تابع میبایست وزن های بهینه و خطای مربع میانگین را به عنوان خروجی بازگرداند.
- برای خطایابی کد خود، می توانید از خروجی کدهای جلسه قبل استفاده کنید. الگوریتم `gradient descent` یا `grid search` را بر روی داده های وزن-قد اجرا کنید و مطمئن شوید که بردار  $w$  خروجی هر سه الگوریتم، یکسان است.

این روش، یک روش مفید برای خطایابی کردن کدهای شماست. به طور مثال، می توانید ابتدا یک روش ساده را پیاده سازی کنید و سپس با استفاده از آن برای چک کردن خروجی توابع پیچیده تر استفاده کنید. اگر هنوز کدهای جلسه قبل را تکمیل نکرده اید، ابتدا آن ها را انجام بدهید.

## ۲.۱ کمترین مربعات همراه با مدل تابع پایه ی خطی

در ادامه، یک مدل تابع پایه را برای دیتایی که در فایل `dataEx3.csv` قرار دارد، پیاده سازی و رسم خواهیم کرد. همانطور که می دانید، رگرسیون خطی ممکن است به طور مستقیم برای داده های غیر خطی، مناسب نباشد. ما از توابع چندجمله ای برای برازش داده های غیر خطی استفاده خواهیم کرد.

$$\phi_j(x) := x^j \quad (1)$$

همانطور که در درس اشاره شد، روش گسترش ویژگی با استفاده از مدل توابع خطی به ما اجازه می دهد که باز هم از روش های رگرسیون خطی برای برازش داده های غیر خطی استفاده کنیم. ( به یاد بیاورید که در تنظیمات اولیه خود، ما فرض می کنیم که هر نقطه ورودی، فقط یک مقدار حقیقی است ) نتیجتاً ما قادر خواهیم بود که داده های خود را با استفاده از درجات مختلف چند جمله ای ها مانند یک چندجمله ای درجه دو ( که ترکیبی از ضرایب  $1, x, x^2$  است ) و یا یک چندجمله ای درجه سه ( که ترکیبی از ضرایب  $1, x, x^2, x^3$  است ) برازش کنیم. درجات بالاتر چندجمله ای از لحاظ محاسبه پیچیده تر خواهند بود اما می توانند جزئیات بیشتری را در داده ها، پوشش دهند که منجر به مدل های رساتری می شود. در مورد برتری ها و کاستی های چندجمله ای های درجه بالاتر و یا پایین تر، بررسی کنید.

برای اندازه گیری میزان متناسب بودن مدل مان، ما از یک تابع هزینه به نام `Root-Mean-Square-Error (RMSE)` استفاده خواهیم کرد. این تابع با استفاده از عبارت زیر به `MSE` مربوط خواهد شد:

$$\text{RMSE}(w) := \sqrt{2 \cdot \text{MSE}(w)} \quad (2)$$

Least Squares<sup>۱</sup>  
overfitting<sup>۲</sup>  
Ridge Regression<sup>۳</sup>  
Linear Basis functions models<sup>۴</sup>

تفسیر اندازه‌ی MSE می‌تواند سخت باشد زیرا این تابع درون خود یک توان‌رسانی دارد. این درحالی است که RMSE اندازه‌گیری تفسیرپذیرتری را بر روی مقیاس مشابهی به مانند خطای یک نقطه، ارائه می‌دهد. به لحاظ خواص آماری، معیارهای بهتری مانند  $R^2$  نیز وجود دارند اما ما فعلاً به آنها نیازی نداریم. اگر به این مبحث علاقه‌مندید، می‌توانید از کتاب Introduction to statistical learning استفاده نمایید. حال می‌خواهیم رگرسیون چندجمله‌ای را با استفاده از تکنیک توابع خطی پیاده سازی کنیم و نتایج پیشبینی آن را مصورسازی کنیم.

## تمرین دوم

هدف این تمرین، نمایش داده‌ها در کنار مقادیر پیشبینی شده با استفاده از رگرسیون چندجمله‌ای است. شما باید  $w$  ای را پیدا کنید که به خوبی هدف ما را نتیجه دهد. این مقدار را باید با استفاده از رگرسیون چندجمله‌ای، به ترتیب با درجات ۱، ۳، ۷ و ۱۳ بدست بیاورید. ممکن است برای این کار نیاز به استفاده از تابعی که در تمرین قبل برای محاسبه RMSE بکار بردید، داشته باشید.

- در فایل این جلسه، تابع `build_poly(x, degree)` را تکمیل کنید. ورودی این تابع یک بردار از نمونه‌های داده  $x_n \in R$ ،  $1 \leq n \leq N$  است. به عنوان خروجی، تابع باید ماتریس ویژگی تعمیم‌یافته‌ی

$$\tilde{\Phi} := \begin{bmatrix} \phi(x_1) \\ \vdots \\ \phi(x_n) \\ \vdots \\ \phi(x_N) \end{bmatrix} \quad \text{where} \quad \phi(x_n) := [1, x_n, x_n^2, x_n^3, \dots, x_n^{\text{degree}}]$$

که ماتریسی است که توسط اعمال توابع چندجمله‌ای بر روی تمامی داده‌های ورودی برای درجات  $j = 0$  تا  $j = \text{degree}$  شکل گرفته است، را برگرداند.

هنگامی که این کار را انجام دادید، باید پیاده‌سازی خود را در فایل جداگانه‌ی `build_polynomial.py` کپی نمایید تا تابع تولید نمودار، بدرستی کار کند.

- اگر کد شما به درستی کار کند، باید داده‌ها و برازش آن را مشاهده کنید و به وضوح خواهید فهمید که چرا رگرسیون خطی یک برازش مناسب نیست، در حالی که رگرسیون چندجمله‌ای یک برازش بهتر تولید می‌کند.
- با تکمیل کردن تابع `polynomial_regression()` خواهید دید که اگر درجه‌ی چندجمله‌ای را افزایش دهیم، مقدار RMSE کاهش می‌یابد. آیا این به معنی بهتر شدن برازش در درجات بالاتر است؟ بنظر شما بهترین برازش چیست؟

## ۲ ارزیابی عملکرد پیشبینی مدل

در عمل، این مهم است که پیشبینی‌ها نه فقط برای داده‌های آموزشی، بلکه برای داده‌های مشاهده نشده نیز خوب عمل کنند. برای شبیه‌سازی حالات واقعی، ما پایگاه داده خود را به دو قسمت تقسیم خواهیم کرد: داده‌های آموزشی، داده‌های آزمون ما با استفاده از نمونه‌های آموزشی، داده‌های خود را برازش خواهیم کرد و RMSE را هم بر روی داده‌های آموزشی و هم بر روی داده‌های آزمون، محاسبه خواهیم نمود.

## تمرین سوم

تابع `train_test_split_demo()` تقسیم داده‌های آموزشی و آزمون را برای درجات مختلف چندجمله‌ای نشان می‌دهد

- برای تقسیم داده‌ها، تابع `split_data(x, y, ratio, ...)` را تکمیل کنید. بنظر شما آیا ترتیب داده‌ها در هنگام جداسازی، اهمیت دارد؟
- تابع `train_test_split_demo()` را کامل کنید. اگر کد شما به درستی کار کند، شما مقدار RMSE را برای درجات ۱، ۳، ۷ و ۱۳ مشاهده خواهید کرد. برای هر درجه، مجدداً سه مقدار RMSE دیگر نیز وجود دارد که مربوط به سه تقسیم بندی داده‌ی زیر است:

- ۹۰٪ آموزش، ۱۰٪ آزمون
- ۵۰٪ آموزش، ۵۰٪ آزمون
- ۱۰٪ آموزش، ۹۰٪ آزمون

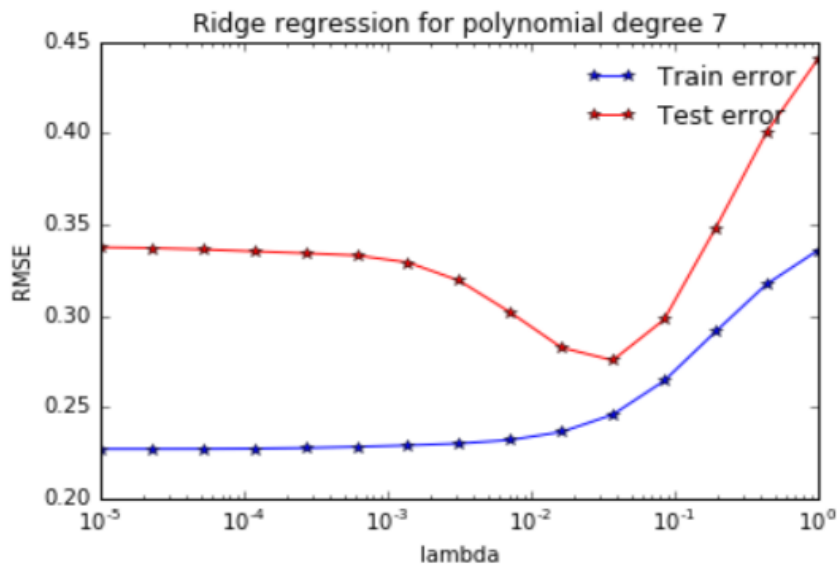
- به مقدار RMSE در داده‌های آموزشی و تست مربوط به درجه‌ی ۳ دقت کنید. آیا قابل تایید و تفسیر است؟ چرا؟
- حال به مقدار RMSE برای دو درجه‌ی دیگر نگاه کنید، آیا قابل تایید و تفسیر هستند؟ چرا؟

- کدام نوع تقسیم، بهتر است؟ چرا؟
- مقدار RMSE برای درجه‌ی 12، به طور مضحکی، برای نوع تقسیم داده‌ی 90% - 10% زیاد است. به نظر شما چرا این اتفاق افتاده است؟ پاسخ را میتوانید با بررسی اشتباهات عددی، پیدا کنید.
- سوال امتیازی: فرض کنید به جای ۵۰ نمونه، ۵۰۰۰ نمونه داشتید، بنظر شما کدام نوع تقسیم در این حالت بهتر است؟

### ۳ رگرسیون ريج

تمرین قبلی، مفهوم بیش برازش را در هنگام استفاده از مدل های پیچیده‌تر نشان می‌دهد. حال میخواهیم با استفاده از رگرسیون ريج این مشکل را حل کنیم.

- در فایل این جلسه، تابع `ridge_regression()` را تکمیل کنید، شما برای خطایابی کد خوتان می‌توانید از  $\lambda = 0$  (ضریب رگولاریزیشن در فرمول ريج) استفاده کنید. با انجام این کار شما نهایتا باید نتیجه‌ای مشابه با کد `least-squares` را در خروجی مشاهده کنید. همچنین می‌توانید این کار را با مقدار بزرگی از  $\lambda$  (مثلا امتحان کنید، RMSE در این حالت بسیار بد خواهد بود).
- در تابع `ridge_regression_demo()` مقدار تقسیم‌بندی خود را برابر پنجاه-پنجاه قرار دهید و مقدار خطای داده‌های آموزشی و آزمون را در مقابل  $\lambda$  برای درجه‌ی ۷، بر روی نمودار نشان دهید. شما باید نموداری مشابه شکل ذیل، را بدست بیاورید:



شکل ۱: ridge regression demo