به نام خداوند بخشنده مهربان

عنوان پروژه

پیش‌بینی قیمت منزل با استفاده از رگرسیون خطی

عنوان درس

یادگیری ماشین

استاد

دکتر الهام قصرالدشتی

دستیاران آموزشی

مهرداد قصابی

مریم صفوی

گردآورنده

سید حسین حسینی

بهار ۱۴۰۴

دانشکده مهندسی کامپیوتر

دانشگاه اصفهان

این پروژه با هدف ساخت یک مدل رگرسیون خطی برای پیش‌بینی یک تابع چندمتغیره‌ی غیرخطی طراحی شده است. ورودی مدل سه متغیر x,y,z بوده و خروجی، مقدار تقریبی تابع چندجمله‌ای $F(x,y,z)$ می‌باشد.

❖ روش تحلیلی

➢ کتابخانه‌ها

- Numpy : برای کار با آرایه‌ها و داده‌ها و انجام عملیات جبری مناسب
- Pandas : برای تحلیل داده‌ها و همچنین انجام عملیات مناسب بر روی دیتافریم‌ها
- Sklearn : یکی از کتابخانه‌های یادگیری ماشین برای انجام عملیات‌های مرتبط با آن
- Gdown : برای دانلود دیتاست از لینک دریافتی

➢ خواندن داده فایل از اکسل

کد ابتدا بررسی می‌کند که آیا فایل وزن‌های آموزش‌یافته وجود دارد یا نه .(regression_weights.npy) در صورت نبود آن، از فایل اکسل (Polynomial_Functions.xlsx) داده‌های آموزشی را می‌خواند.

➢ تولید بردار ویژگی ها

توابعی برای استخراج ویژگی‌ها از داده‌های خام پیاده‌سازی شده است، در راستای اینکه معادله مورد نظر به ما داده شده و ما هم با استفاده از مهندسی ویژگی، ویژگی های مورد نیاز را بدست می آوریم تا بتوانیم ماتریس ویژگی ها را ساخته و مقدار وزن ها را مشخص کنیم.

این تابع ۲۰ ویژگی برای ترکیب‌های چندجمله‌ای مختلف از سه متغیر تولید می‌کند. این ویژگی‌ها شامل توانی از متغیرها و ترکیب‌های آن‌ها مثل x^2y, xyz هستند.

➢ آموزش مدل با معادله نرمال

با استفاده از معادله نرمال زیر وزن‌ها آموزش داده می‌شوند:

$$\beta = (X^TX)^{-1}X^TY$$

در صورتی که ماتریس منفرد باشد، از شبه‌وارون (Pseudo-Inverse) استفاده می‌شود.

➢ ذخیره سازی یا بارگذاری وزن ها

در صورت موفقیت‌آمیز بودن آموزش، وزن‌ها در فایل npy.ذخیره می‌شوند و در اجراهای بعدی مستقیماً بارگذاری خواهند شد و با توجه به همین مسئله وزن های بدست آمده را میتوانیم در مدل استفاده کنیم.

❖ روش گرادیان کاهشی

معادله هدف مدل:

$$F(x,y,z) = b + w_1x + w_2x^2 + w_3x^3 + w_4y^1 + w_5y^2 + w_6y^3 + w_7z + w_8z^2 + w_9z^3 + w_{10}xy + w_{11}x^2y + w_{12}xy^2 + w_{13}xz + w_{14}x^2z + w_{15}xz^2 + w_{16}yz + w_{17}y^2z + w_{18}yz^2 + w_{19}xyz$$

که در آن b بایاس (bias) و w_i وزن‌های (weights) مربوط به هر ویژگی هستند.

➢ کتابخانه‌ها

- pandas برای کار با داده‌ها خواندن فایل اکسل، ایجاد.
- numpy برای محاسبات عددی (کار با آرایه‌ها، عملیات ریاضی).
- scikit-learn برای تقسیم داده .(train_test_split) توجه: مدل اصلی با گرادیان کاهشی دستی پیاده‌سازی شده، نه با LinearRegression اسکیت‌لرن برای آموزش نهایی.
- Matplotlib برای رسم نمودارها (نمودار یادگیری).
- seaborn برای بهبود ظاهر نمودارها (اختیاری).
- gdown برای دانلود فایل از گوگل درایو در این کد استفاده نشده، داده از فایل محلی خوانده می‌شود.
- warnings برای مدیریت هشدارهای پایتون.

فایل داده :فایل اکسل با نام Polynomial_Functions.xlsxباید در مسیر اجرای نوت‌بوک وجود داشته باشد یا مسیر صحیح آن در کد مشخص شود.

➢ شرح دیتاست(Dataset Description)

- ساختار :دیتاست شامل ۱۰۰۰۰ نمونه (ردیف) و ۴ ستون است:
  - x:ویژگی ورودی اول.(float64)
  - y:ویژگی ورودی دوم.(float64)
  - z:ویژگی ورودی سوم.(float64)
  - F(x, y, z):متغیر هدف یا خروجی.(float64)
- کیفیت داده :داده‌ها فاقد مقادیر گمشده (Missing Values) هستند و همگی از نوع عددی می‌باشند. آمار توصیفی و تعداد مقادیر یکتا در کد بررسی شده است.

➢ مراحل پیاده‌سازی و متدولوژی

پروژه مراحل زیر را دنبال می‌کند:

1. بارگذاری و بررسی داده:
   - فایل اکسل با استفاده از pandas.read_excel خوانده می‌شود.
   - اطلاعات کلی دیتاست(df.info()) ، مقادیر گمشده(df.isnull().sum()) ، آمار توصیفی (df.describe()) و تعداد مقادیر یکتا (df.nunique()) نمایش داده می‌شود.
2. تقسیم داده:(Train/Test Split)
   - داده‌ها به دو بخش آموزش (Train) و آزمون (Test) تقسیم می‌شوند.
   - ویژگی‌ها (x, y, z) در متغیر Xو متغیر هدف (F(x, y, z)) در متغیر yقرار می‌گیرند.
   - از train_test_splitبا نسبت 80٪ برای آموزش و 20٪ برای آزمون (test_size=0.2) و random_state=42برای تکرارپذیری نتایج استفاده می‌شود.
3. پیش‌پردازش:(Preprocessing)
   - حذف داده‌های پرت:(Remove Outliers)
     - تابعی به نام remove_outliers_auto_xy_with_outputتعریف شده که از روش دامنه بین چارکی (IQR - Interquartile Range) برای شناسایی و حذف داده‌های پرت استفاده می‌کند.
     - این تابع *فقط روی داده‌های آموزش* (X_train, y_train)اعمال می‌شود.
     - نکته: بر اساس خروجی کد، در این اجرا فقط داده‌های پرت مربوط به متغیر هدف (y) حذف شده‌اند.
   - نرمال‌سازی داده:(Normalize Data)
     - تابعی به نام normalize_dataتعریف شده که نرمال‌سازی Z-score (میانگین صفر و انحراف معیار یک) را انجام می‌دهد.
     - میانگین و انحراف معیار *فقط از داده‌های آموزش* محاسبه می‌شود.
     - این مقیاس‌بندی روی هر دو مجموعه داده آموزش و آزمون (هم ویژگی‌ها Xو هم هدف (yاعمال می‌شود.
     - پارامترهای نرمال‌سازی (میانگین و انحراف معیار) برای ویژگی‌ها (feature_scaler) و هدف (output_scaler)ذخیره می‌شوند تا بعداً برای پیش‌بینی ورودی جدید و دنرمال‌سازی خروجی استفاده شوند.
     - تابعی برای دنرمال‌سازی (denormalize) نیز تعریف شده است.

4. مهندسی ویژگی:(Feature Engineering)
   - تابعی به نام feature_engineeringتعریف شده است.
   - این تابع ورودی‌های نرمال‌شده x, y, zرا گرفته و ۱۹ ویژگی جدید مطابق با معادله چندجمله‌ای مورد نظر ایجاد می‌کند:

$$x, x^2, x^3, y, y^2, y^3, z, z^2, z^3, xy, x^2y, xy^2, xz, x^2z, xz^2, yz, y^2z, yz^2, xyz$$

   - این تبدیل روی داده‌های نرمال‌شده آموزش (X_train_norm) و آزمون (X_test_norm) اعمال می‌شود و بردارهای ویژگی گسترش‌یافته (X_train_vector, X_test_vector) را تولید می‌کند. این کار باعث می‌شود مدل رگرسیون خطی بتواند روابط غیرخطی را مدل کند.
5. پیاده‌سازی و آموزش مدل:(Model Implementation)
   - تابع هزینه، خطای میانگین مربعات (mse_loss) به صورت دستی تعریف می‌شود.
   - تابع linear_regressionبرای آموزش مدل با استفاده از گرادیان کاهشی پیاده‌سازی شده است.
   - این تابع وزن‌ها (weights) و بایاس (bias) را به صورت تکراری در طول epochsبا استفاده از learning_rateمشخص شده، به‌روزرسانی می‌کند.
   - ورودی‌های این تابع، بردارهای ویژگی مهندسی‌شده (X_train_vector, X_test_vector) و مقادیر هدف نرمال‌شده (y_train_norm, y_test_norm) هستند.
   - در هر epoch ، خطای آموزش و آزمون محاسبه و چاپ می‌شود.
   - در نهایت، وزن‌ها و بایاس نهایی مدل و همچنین مقادیر خطای نهایی آموزش و آزمون برگردانده می‌شوند.
   - نمودار یادگیری (Learning Curve) که خطای آموزش و آزمون را در طول epoch ها نشان می‌دهد، با استفاده از matplotlibرسم می‌شود تا روند همگرایی مدل بررسی شود.
6. پیش‌بینی با تابع چندجمله‌ای:(Polynomial Function Prediction)
   - آخرین بخش کد به کاربر اجازه می‌دهد مقادیر جدیدی برای x, y, zوارد کند.
   - مراحل زیر برای هر ورودی جدید انجام می‌شود:
     1. ورودی کاربر (x, y, z) دریافت می‌شود.
     2. ورودی‌ها با استفاده از (feature_scalerمقیاس کننده ویژگی‌ها که در مرحله پیش‌پردازش ذخیره شد (نرمال‌سازی می‌شوند.
     3. ویژگی‌های چندجمله‌ای و تعاملی با استفاده از تابع feature_engineeringروی ورودی نرمال‌شده ساخته می‌شوند.

4. پیش‌بینی نرمال‌شده با استفاده از وزن‌ها و بایاس یادگرفته شده محاسبه می‌شود (np.dot(input_norm_vector, weights) + bias).
5. نتیجه پیش‌بینی نرمال‌شده با استفاده از output_scaler (مقیاس‌کننده هدف (دنرمال‌سازی می‌شود تا به مقیاس اصلی تابع F برگردد.
6. مقدار پیش‌بینی نهایی (دنرمال‌شده) چاپ می‌شود.

۵ .توضیح توابع کلیدی

- remove_outliers_auto_xy_with_output(x, y): داده‌های پرت را با روش IQR از x و y (داده آموزش) حذف می‌کند.
- normalize_data(X_train, X_test, y_train, y_test): نرمال‌سازی Z-score را روی داده‌های آموزش و آزمون اعمال کرده و مقیاس‌کننده‌ها را برمی‌گرداند.
- denormalize(data, scaler): داده نرمال‌شده را با استفاده از مقیاس‌کننده به مقیاس اصلی برمی‌گرداند.
- feature_engineering(df): ویژگی‌های چندجمله‌ای و تعاملی درجه ۳ را از ستون‌های 'x', 'y', 'z' یک DataFrame می‌سازد.
- mse_loss(y_true, y_pred): خطای میانگین مربعات را محاسبه می‌کند.
- linear_regression(X_train, y_train, X_test, y_test, learning_rate, epochs): مدل رگرسیون خطی را با گرادیان کاهشی دستی روی داده‌های ورودی آموزش می‌دهد و وزن‌ها، بایاس و تاریخچه خطا را برمی‌گرداند.

۶ .نحوه استفاده

1. نصب پیش‌نیازها :اطمینان حاصل کنید که تمام کتابخانه‌های لیست شده در بخش ۲ نصب هستند (pip install pandas numpy scikit-learn matplotlib seaborn gdown openpyxl).
2. آماده‌سازی داده :فایل Polynomial_Functions.xlsx را در کنار فایل نوت‌بوک قرار دهید یا مسیر آن را در کد (سلول مربوط به pd.read_excel) به‌روز کنید.
3. اجرای نوت‌بوک :سلول‌های نوت‌بوک را به ترتیب از بالا به پایین اجرا کنید.
4. مشاهده نتایج :خروجی هر سلول را بررسی کنید، از جمله اطلاعات دیتاست، اندازه‌های مجموعه آموزش/آزمون، روند حذف داده‌های پرت، داده‌های نرمال‌شده، بردارهای ویژگی مهندسی‌شده، روند کاهش خطا در طول آموزش (خروجی متنی و نمودار یادگیری) و وزن‌ها و بایاس نهایی مدل.
5. انجام پیش‌بینی :در آخرین سلول، وقتی برنامه از شما مقادیر x, y, و z را درخواست کرد، اعداد مورد نظر خود را وارد کنید تا پیش‌بینی مدل برای آن ورودی نمایش داده شود.

۷ .نتایج

- مدل با موفقیت بر روی داده‌های مهندسی‌شده آموزش داده شده است.
- نمودار یادگیری نشان می‌دهد که خطای آموزش و آزمون در طول زمان کاهش یافته و مدل به همگرایی رسیده است.
- مقادیر نهایی خطای میانگین مربعات (MSE) برای داده‌های نرمال‌شده آموزش و آزمون بسیار پایین و نزدیک به هم هستند (حدود 0.005)، که نشان‌دهنده برازش (fit) خوب مدل بدون بیش‌برازش (overfitting) قابل توجه است.
- وزن‌ها و بایاس نهایی مدل که ضرایب معادله چندجمله‌ای را نشان می‌دهند، محاسبه و چاپ شده‌اند.
- بخش نهایی کد امکان پیش‌بینی مقادیر جدید Fرا برای ورودی‌های دلخواه x, y, zفراهم می‌کند.

۸ .نکات مهم

- حذف داده‌های پرت و محاسبه پارامترهای نرمال‌سازی (میانگین و انحراف معیار) *فقط* بر اساس داده‌های آموزش انجام می‌شود تا از نشت اطلاعات از مجموعه آزمون به آموزش (Data Leakage) جلوگیری شود.
- مدل رگرسیون با استفاده از گرادیان کاهشی دستی پیاده‌سازی شده است که درک عمیق‌تری از نحوه کارکرد الگوریتم ارائه می‌دهد.

➢ منابع

- Ng, Andrew. Machine Learning (Coursera)
- دوره یادگیری ماشین، دانشگاه صنعتی شریف .شریفی زارچی، علی
- Géron, Aurélien. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow. O'Reilly Media, 2019
- OpenAI. ChatGPT

سید حسین حسینی دولت آبادی

موفق باشید 😊😍