TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH

KHOA ĐÀO TẠO CHẤT LƯỢNG CAO

NGÀNH CÔNG NGHỆ THÔNG TIN

🙡🙣



BÁO CÁO ĐỒ ÁN 2

GVHD: TS. Trần Nhật Quang

SVTH: Nguyễn Việt Hoàng 16110076

Trần Văn Việt 16110265

Lớp: 16110ST3

MỤC LỤC

[I. Tìm hiểu về đạo văn và lời cam kết 1](#_Toc9528642)

[**1.** **Tìm hiểu về đạo văn** 1](#_Toc9528643)

[**2.** **Lời cam kết** 1](#_Toc9528644)

[II. Tìm hiểu lý thuyết 1](#_Toc9528645)

[**1.** **Ridge Regression** 2](#_Toc9528646)

[**2.** **Lasso Regression** 3](#_Toc9528647)

[**3.** **Tổng quan so sánh giữa 2 thuật toán[5]** 3](#_Toc9528648)

[III. Dataset[7] 4](#_Toc9528649)

[**1.** **Mô tả(House\_price.csv)** 4](#_Toc9528650)

[**2.** **Thuộc tính** 4](#_Toc9528651)

[IV. Giới thiệu về thư viện Sklearn 6](#_Toc9528652)

[**1.** **Giới thiệu** 6](#_Toc9528653)

[**2.** **Thư viện Sklearn đối mỗi thuật toán** 7](#_Toc9528654)

[V. Đánh giá hiệu năng 8](#_Toc9528655)

[**1.** **Lý thuyết[8]** 8](#_Toc9528656)

[**2.** **Lasso Regression và Ridge Regression** 9](#_Toc9528657)

[VI. TÀI LIỆU THAM KHẢO 11](#_Toc9528658)

# Tìm hiểu về đạo văn và lời cam kết

1. **Tìm hiểu về đạo văn**
2. **Định nghĩa[1]**

Đạo văn là "chiếm hữu một cách sai trái" và "ăn cắp và công bố" "ngôn ngữ, suy nghĩ, ý tưởng, hay cách diễn đạt" của người khác và xem chúng như là những gì do mình tự tạo ra.

Đạo văn được xem là hành vi thiếu trung thực về mặt học thuật và vi phạm đạo đức báo chí. Người nào đạo văn sẽ bị phạt tiền, bị đình chỉ, và thậm chí bị đuổi học hay đuổi việc.

Trong môi trường học thuật và công việc, đạo văn là một hành vi vi phạm đạo đức rất nghiêm trọng; một số trường hợp đạo văn có thể cấu thành hành vi vi phạm bản quyền.

1. **Cách phòng tránh[2]**

* **Những điều nên làm:** 
  + Nếu bắt buộc phải sao chép thì ít nhất bạn hãy cố gắng hiểu và biến tấu thông tin theo văn phong của bản thân và để phần đã sao chép trong dấu trích dẫn (ngoặc đơn hoặc ngoặc kép) chứ **đừng sao chép cả trang hoặc cả đoạn văn**. Sau đó, trích dẫn nguồn mà bạn đã tham khảo theo định dạng tài liệu tham khảo phù hợp.
* Dùng từ đồng nghĩa với từ mình muốn dùng.
* Thay đổi cách dạng câu văn.
* Trích nguồn ngay sau khi bạn trích dẫn. Tránh để sau vì có thể bạn sẽ quên một hoặc một vài trích dẫn và điều đó sẽ trở thành đạo văn.
* **Những điều không nên làm:**
* Việc nhờ ai đó sửa bài viết của bạn bằng cách thêm bớt một vài từ hoặc câu văn vẫn coi là nhờ người đó làm hộ một phần việc của bạn, và được xem như là hành vi đạo văn.
* Không ghi nguồn.
* Đừng cố gắng ăn cắp chất xám của người khác bằng việc đạo văn.

1. **Lời cam kết**

“Chúng tôi xin cam đoan đồ án này do chính chúng tôi thực hiện. Chúng tôi không sao chép, sử dụng bất kỳ tài liệu, mã nguồn của người khác mà không ghi rõ nguồn gốc. Chúng tôi xin chịu hoàn toàn trách nhiệm nếu vi phạm.”.

# Tìm hiểu lý thuyết

Trước khi bắt đầu, chúng ta nên tự đặt câu hỏi rằng khi nào thì ta biết mô hình của mình đang bị underfitting hay overfitting trước khi nghĩ đến chuyện giải quyết chúng khi nào.

* Underfiting đơn thuần chỉ là mô hình của chúng ta quá đơn giản, nên ta cần thêm những thành phần đa thức để nó phức tạp hơn. Nên khi giá trị của cost function lớn ta sẽ đẩy bậc của hàm số lên và sẽ khiến chúng ta gặp phải những vấn đề khác về performance và overfitting.
* Với Overfitting thì ngược lại, do mô hình quá phức tạp nên ta cần giảm bậc của hàm số hay số lượng feature. Việc giảm feature ngoài bỏ bớt những thành phần đa thức, chúng ta còn bỏ đi được những feature không cần thiết. Một cách khác là áp dụng Regularization.

Vậy Regularization là gì? Nó là một công nghệ giúp giảm độ phức tạp của mô hình. Nó làm được như vậy bởi vì nó thêm một biến số vào phía trong hàm cost function. Nó sẽ giúp giải quyết hiện tượng Overfitting. Cụ thể hơn của Regularization, chúng ta có Ridge Regression và Lasso Regression. Hai nội dung sẽ được trình bày ở dưới.

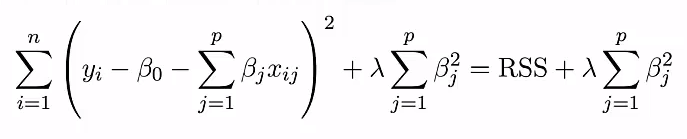
1. **Ridge Regression**
2. **Ứng dụng**

Được dùng trong phân tích các kháng nguyên đặc hiệu tuyến tiền liệt và các biện pháp lâm sàng giữa những người sắp cắt bỏ tuyến tiền liệt cả họ[4]

1. **Mô tả thuật toán[5]**

Là biến thể của hồi quy tuyến tính

Chúng ta có công thức hồi quy Ridge tối thiểu hóa như sau:



Từ công thức tối thiểu hóa tổng phần dư bình phương(RSS) của hồi quy tuyến tính, chúng ta cộng thêm vào RSS tổng ()2 nhân với hệ số lambda. Hệ số lambda là hệ số tùy chỉnh được ước lượng trước khi khớp với mô hình. Và nó dùng để ước lượng giữa hai đại lượng:

* Nếu như lambda lớn thì hầu hết các hế số β sẽ giảm dần về 0 dẫn đến hiện tượng underfitting
* Nếu như lambda bằng không thì thực chất mô hình sẽ trở về dạng hồi quy thông thưởng
* Chính vì vậy mà việc xác định lambda là rất quan trọng và người ta thường sử dụng phương pháp cross-validation

Có thể thấy thì Ridge Regression cũng tương tự như hồi quy tuyến tính nhằm làm tối thiểu hóa RSS tuy nhiên chúng ta có thêm biểu thức: tổng ()2 nhân với hệ số lambda. Giá trị biểu thức lớn nếu như hệ số β lớn, hồi quy Ridge sẽ ưu tiên mô hình với giá trị β nhỏ hơn. Do đó các biến không có ý nghĩa sẽ giảm gần về bằng 0.

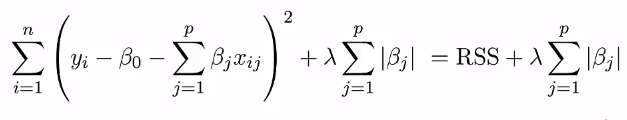
1. **Lasso Regression**
2. **Ứng dụng[6]**

Được sử dụng rộng rãi trong mạng lưới tài chính và kinh tế. Trong kinh tế, nó được sử dụng trong dự báo xác suất mặc định và được sử dụng trong dự báo rủi ro cho toàn doanh nghiệp.

1. **Mô tả thuật toán[5]**

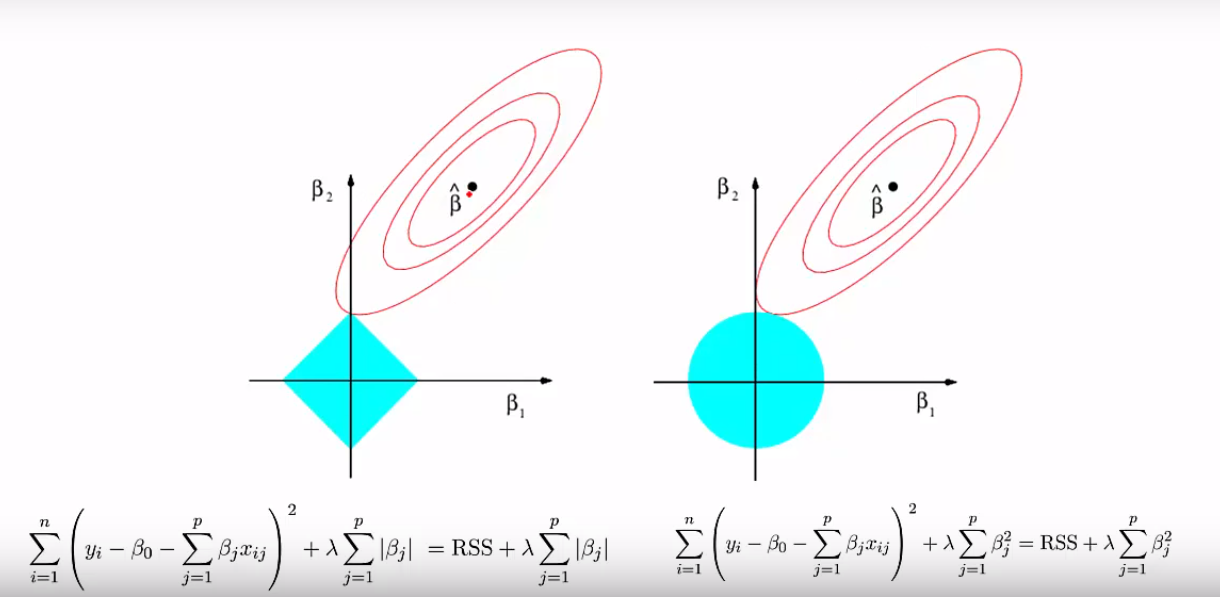
Đối với Ridge Regression chúng ta vừa tìm hiểu ở trên, nó sẽ giữ lại tất cả các biến số, chỉ làm giảm hệ số của chúng gần xấp xỉ 0. Tuy nhiên chúng ta có một thuật toán khác giúp chọn lựa các biến có ý nghĩa, còn các biến không có ý nghĩa sẽ được gán giá trị bằng 0(có nghĩa là các biến số này sẽ bị loại bỏ hoàn toàn ra khỏi mô hình).

Ta có công thức hồi quy Lasso tối thiểu hóa như sau:



Theo như công thức phía trên, chúng ta có thể thấy rằng nó gần giống so với hồi quy Ridge. Hồi quy Lasso sẽ tính tổng giá trị tuyệt đối của thay vì tính tổng giá trị bình phương của trong hồi quy Ridge.

1. **Tổng quan so sánh giữa 2 thuật toán[5]**



Phía bên trái là Lasso Regression và phía bên phải là Ridge Regression.

Hình thoi màu xanh có ý nghĩa là tập hợp các điểm có tổng giá trị tuyệt đối của trong hồi quy Lasso và hình tròn màu xanh có ý nghĩa là tập hợp các điểm có tổng giá trị bình phương của trong hồi quy Ridge. Còn các đường viền elip được biểu diện cho RSS. Ta thấy rằng cả hai đều tìm được các hệ số khi mà đường viền elip tiếp xúc với vùng tập hợp các điểm của hai hồi quy. Tuy nhiên ta thấy rằng, đồi với hồi quy Lasso, các góc của tập điểm nằm trên trục tọa độ. Điều đó chứng tỏ rằng, khi mà đường viền elip tiếp xúc với những điểm này thì feature sẽ không được sử dụng và hệ số được gán bằng 0. Còn đối với hồi quy Ridge thì ta có thể thấy rằng khi các đường viền elip chạm vào tập điểm thì các hệ số chỉ có thể tiến gần về tới 0 chứ không thể bằng 0 như hồi quy Lasso.

# Dataset[7]

1. **Mô tả(House\_price.csv)**

Dataset gồm 1460 hàng và 81 cột dự đoán giá bán nhà dựa vào một vài thông tin như: đường, nơi ở, khu vực dân cư, chất lượng căn nhà, điều kiện bán, …

1. **Thuộc tính**

* ID: số thứ tự
* MSSubClass: xác định loại nhà ở liên quan đến bán hàng
* MSZoning:xác định phân loại quy hoạch chung về việc bán hàng:

|  |  |
| --- | --- |
| Ký hiệu | Nội dung |
| A | Nông nghiệp |
| C | Thương mại |
| FV | Khu dân cư |
| I | Công nghiệp |
| RH | Mật độ dân cư cao |
| RL | Mật độ dân cư thấp |
| RP | Khu dân cư có mật độ thấp |
| RM | Mật độ trung bình dân cư |

* LotFrontage: Chân tuyến tính của đường phố kết nối với tài sản
* LotArea: kích thước lô tinh bằng feet
* Street: Loại đường
* LotShape: Hình dạng chung của tài sản

|  |  |
| --- | --- |
| Ký hiệu | Nội dung |
| Reg | Đều đặn |
| IR1 | Hơi bất thường |
| IR2 | Bất thường vừa phải |
| IR3 | Bất thường |

* LandContour: Độ phẳng của tài sản
* Utilities: Loại tiện ích có sẵn
* LotConfig: cấu hình lô
* LandSlope: độ dốc của tài sản
* Neighborhood: địa điểm thực tế trong thành phố
* Condition1: các điều kiện 1
* Condition2: các điều kiện 2
* BldgType: loại nhà ở
* HouseStyle: phong cách nhà ở
* OverallQual: đánh giá vật liệu của căn nhà
* OverallCond: đánh giá tình trạng chung của ngôi nhà
* YearBuilt: ngày thi công ban đầu
* YearRemodAdd: ngày tu sửa
* RoofStyle: vật liệu mái
* Exterior1st: ngoại thất
* Exterior2nd: ngoại thất bao phủ căn nhà
* MasVnrType: loại vật liệu xây nhà
* MasVnrArea: diện tích vật liệu (feet)
* ExterQual: đánh giá chất lượng vật liệu bên ngoài
* ExterCond: đánh giá tình trạng vật liệu hiện tại
* Foundation: Loại móng
* BsmtQual: đánh giá chiều cao của tầng hầm
* BsmtCond: đánh giá tình trạng chung của tầng hầm
* BsmtExposure: đề cập đến nối đi hoặc đường trong vườn
* BsmtFinType1: đánh giá diện tích tầng hầm
* BsmtFinSF1: diện tích tầng hầm
* BsmtFinType2: đánh giá diện tích tầng hầm loại 2
* BsmtFinSF2:
* BsmtUnfSF: tầng hầm chưa hoàn thành
* TotalBsmtSF: tổng diện tích tầng hầm
* Heating: Loại lò sưởi ấm
* HeatingQC: chất lượng của lò sưởi ấm
* CentralAir: điều hòa
* Electrical: hệ thống điện
* 1stFlrSF: diện tích tầng 1
* 2ndFlrSF: diện tích tầng 2
* LowQualFinSF: chất lượng hoàn thành cấc tầng
* GrLivArea: diện tích tầng trên
* BsmtFullBath: tầng hầm đầy đủ phòng tắm
* BsmtHalfBath: tầng hầm có gần đầy đủ phòng tắm
* FullBath: tầng trên có đầy đủ phòng tắm
* HalfBath: tầng trên có gần đầy đủ phòng tắm
* Bedroom: phòng ngủ ở tầng trên
* Kitchen: phòng bếp ở tầng trên
* KitchenQual: chất lượng phòng bếp
* TotRmsAbvGrd: tổng số phòng ở tầng trên
* Functional: chức năng của căn nhà
* Fireplaces: số bếp nấu
* FireplaceQu: chất lượng bếp nấu
* GarageType: vị trí gara
* GarageYrBlt: năm xây gara
* GarageFinish: đã hoàn thành gara hay chưa
* GarageCars: số xe của gara có thể chứa
* GarageArea: số diện tích gara
* GarageQual: chất lượng của gara
* GarageCond: điều kiện để nhà xe
* PavedDrive: đường lái xe
* WoodDeckSF: diện tích sàn gỗ
* OpenPorchSF: diện tích hiên mở
* EnclosedPorch
* 3SsnPorch:
* ScreenPorch: màn hình diện tích bằng met vuông
* PoolArea: diện tích hồ bơi
* PoolQC: chất lượng hồ bơi
* Fence: chất lượng hàng rào
* MiscFeature
* MiscVal: giá trị của các tính nagw
* MoSold: tháng bán
* YrSold: năm bán
* SaleType: loại bán
* SaleCondition:điều kiện bán
* Saleprice: giá bán

# Giới thiệu về thư viện Sklearn

1. **Giới thiệu**

Viết tắt là sklearn, là một thư viện mã nguồn mở trong ngành Machine Learning, rất mạnh mẽ và phổ biến trong cộng đồng python, được thiết kế trên nền tảng NumPy và SciPy.

Sklearn chứa hầu hết các thuật toán Machine Learning hiện đại nhất, đi kèm với comprehensivevà documents. Điểm mạnh của thư viện này là nó được sử dụng phổ biến trong academia và industry, do đó nó luôn được update và có một very user community.

1. **Thư viện Sklearn đối mỗi thuật toán**
2. **Ridge Regression**

#gan du lieu x y

x = df.iloc[:,:80]

y = df.iloc[:,-1]

#chia 80% du lieu de train va 10% de test

x\_train, x\_test, y\_train, y\_test = train\_test\_split(x, y, test\_size=0.1,random\_state=2)

#train du lieu

#mac dinh anpha =1

ride = Ridge(alpha=1)

ride.fit(x\_train,y\_train)

y\_pre=ride.predict(x\_test)

print('anpha=1\nscore: ',r2\_score(y\_test,y\_pre))

#anpha =0.01

ride\_001 = Ridge(alpha=0.01)

ride\_001.fit(x\_train,y\_train)

y\_pre001=ride\_001.predict(x\_test)

print('anpha=0.01\n score: ',r2\_score(y\_test,y\_pre001))

#anpha = 0.001

ride\_0001 = Ridge(alpha=0.001)

ride\_0001.fit(x\_train,y\_train)

y\_pre001=ride\_0001.predict(x\_test)

print('anpha=0.001\n score: ',r2\_score(y\_test,y\_pre001))

#anpha = 0.0001

ride\_00001 = Ridge(alpha=0.0001)

ride\_00001.fit(x\_train,y\_train)

y\_pre001=ride\_00001.predict(x\_test)

print('anpha=0.0001\n score: ',r2\_score(y\_test,y\_pre001))

1. **Lasso Regression**

#gan du lieu x y

x = df.iloc[:,:80]

y = df.iloc[:,-1]

#chia 90% du lieu de train va 10% de test

x\_train, x\_test, y\_train, y\_test = train\_test\_split(x, y, test\_size=0.1,random\_state=2)

#train du lieu

#mac dinh anpha =1

lasso = Lasso()

lasso.fit(x\_train,y\_train)

y\_pre=lasso.predict(x\_test)

print('anpha=1\nscore: ',r2\_score(y\_test,y\_pre))

coeff\_1=np.sum(lasso.coef\_!=0)

print('tong coeff khac khong',coeff\_1)

#anpha =0.01

lasso\_001 = Lasso(alpha=0.01)

lasso\_001.fit(x\_train,y\_train)

y\_pre001=lasso.predict(x\_test)

print('anpha=0.01\n score: ',r2\_score(y\_test,y\_pre001))

#tinh tong he so khac khong

coeff\_001=np.sum(lasso\_001.coef\_!=0)

print('tong coeff khac khong',coeff\_001)

#anpha = 0.001

lasso\_001 = Lasso(alpha=0.001)

lasso\_001.fit(x\_train,y\_train)

y\_pre001=lasso.predict(x\_test)

print('anpha=0.001\n score: ',r2\_score(y\_test,y\_pre001))

#tinh tong he so khac khong

coeff\_001=np.sum(lasso\_001.coef\_!=0)

print('tong coeff khac khong',coeff\_001)

#anpha = 0.0001

lasso\_001 = Lasso(alpha=0.0001)

lasso\_001.fit(x\_train,y\_train)

y\_pre001=lasso.predict(x\_test)

print('anpha=0.0001\n score: ',r2\_score(y\_test,y\_pre001))

#tinh tong he so khac khong

coeff\_001=np.sum(lasso\_001.coef\_!=0)

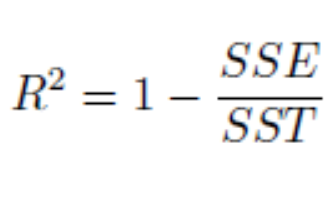
print('tong coeff khac khong',coeff\_001)

# Đánh giá hiệu năng

1. **Lý thuyết[8]**

R2 (Hệ số xác định)

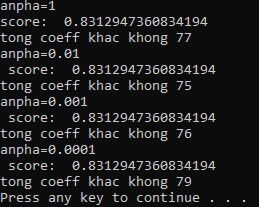
* Luôn nằm trong khoảng 0 đến 1 hoặc 0% đến 100%, giá trị càng gần 1 thì sự dự đoán của mô hình càng đúng



* Trong đó:
* SSE: (Tổng bình phương lỗi) Tổng bình phương giá trị thực tế trừ giá trị tìm được
* SST(Tổng bình phương của phần dư) Tổng bình phương giá trị thực tế - giá trị trung bình

1. **Lasso Regression và Ridge Regression**

Output đối với lasso:

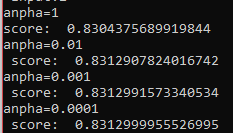


Như output ở trên ta có thể thấy với:

* anpha bằng 1 thì score =0.83, với 80 features nhưng chỉ sử dụng 77 features còn 3 features bằng 0
* anpha bằng 0.01 thì score =0.83, với 80 features nhưng chỉ sử dụng 75 features còn 5 features bằng 0
* anpha bằng 0.001 thì score =0.83, với 80 features nhưng chỉ sử dụng 76 features còn 4 features bằng 0
* anpha bằng 0.0001 thì score =0.83, với 80 features nhưng chỉ sử dụng 79 features còn 1 features bằng 0

theo ta thấy số features mà lasso cho về 2 phụ thuộc vào anpha truyền vào ta phải thử mọi trường hợp anpha sao cho là hợp lý nhất.

Output đối với Ridge:



Như output ở trên ta có thể thấy:

Với ridge thì nó không làm cho các features về 0 nên không thể xem được, nhưng đối với từng anpha thì nó cho ra từng score khác nhau tốt nhất là anpha= 0.0001

Với 2 score tốt nhất của 2 mô hình thì không chênh lệch nhau nhiều lắm nhưng với dataset này thì score bên ridge tốt hơp nên nó hợp với ridge hơn

# TÀI LIỆU THAM KHẢO

[1] Đạo văn – Wikipedia tiếng Việt. (2005, June 22). Retrieved May 19, 2019, from https://vi.wikipedia.org/wiki/%C4%90%E1%BA%A1o\_v%C4%83n

[2] (2016, May 8). Retrieved from <https://www.wikihow.vn/Tr%C3%A1nh-%C4%90%E1%BA%A1o-V%C4%83n>

[4]: Ridge Regression. (2017, October 5). Retrieved from <https://mindmajix.com/ridge-regression>

[5]: Machine learning: H'i quy Ridge và h'i quy LASSO. (2017, July 16). Retrieved from <https://www.youtube.com/watch?v=XEo7cV-rfnw&t=115s>

[6]: Top 6 Regression Algorithms Used In Data Mining And Their Applications In Industry. (2019, May 12). Retrieved from <https://www.analyticsindiamag.com/top-6-regression-algorithms-used-data-mining-applications-industry/>

[7]: Ridge Regression | Kaggle. (n.d.). Retrieved from <https://www.kaggle.com/alaaawad/ridge-regression/data>

[8]: Khandelwal, R. (2018, November 1). How to evaluate the performance of a machine learning model. Retrieved from https://medium.com/datadriveninvestor/how-to-evaluate-the-performance-of-a-machine-learning-model-45063a7a38a7