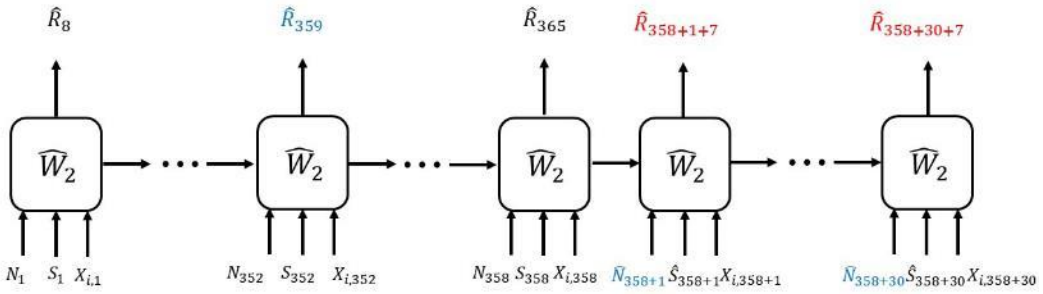
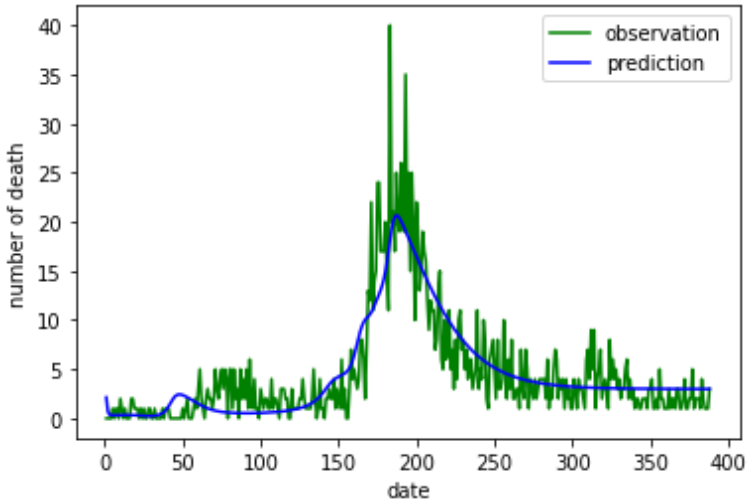
Neural Forecasting of the Mortality of COVID-19 using LSTM Model

이화여자대학교 석사학위 논문 (2022.02 졸업)

#Python #COVID-19 #Deep Learning #Machine Learning





- 1년치(2020.06.30 ~ 2021.06.30) **코로나19 관련 데이터를** 학습하여 한 달치(2021.07.01 ~ 2021.07.30) 코로나19 감염자 수와 사망자 수를 예측하는 모델을 개발하였습니다.
- 순차적인 입력 데이터 간의 거리가 멀어도 잘 기억하고 학습하는 **LSTM을** 딥러닝 알고리즘으로 사용했습니다.
- **설명변수**로는 우리나라의 확진자 연령층과 백신 접종률, 4가지 변이바이러스(알파, 베타, 감마, 델타)의 감염률을 적용하고 잠복기간은 논문 작성 당시 최소기간으로 발표된 7일로 설정하였습니다.
- 일일 예측치를 시각화 하였을 때 확진자 수나 사망자 수가 급증한 구간은 약간의 오차를 보이나 RMSE가 3으로 전반적으로 **높은 예측력을** 보이고 있습니다.

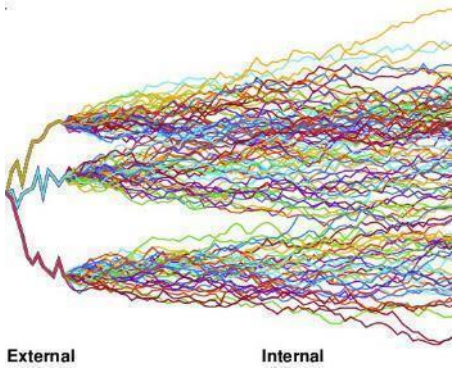
고유정

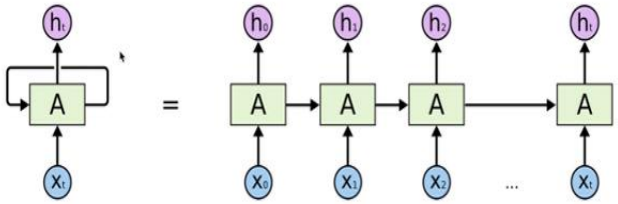
자동차사고 데이터를 활용한 SSM 기반 사고율 예측과 성능비교

금융위험관리 최종 프로젝트 (2021.04 ~ 2021.06)

#Python #R #Monte Carlo Simulation #Deep Learning #Machine Learning #Tensorflow #Pytorch

[Nested Monte Carlo Simulation]





[RNN-LSMC Simulation]

```
model_many_to_one = keras.models.Sequential([
  keras.layers.LSTM(10, return_sequences=True, input_shape=[None, 2]),
  keras.layers.LSTM(10, return_sequences=True),
  keras.layers.LSTM(10, return_sequences=True),
  keras.layers.LSTM(10, return_sequences=False),
  keras.layers.Dense(1, activation=tf.keras.activations.exponential)
  # complete here
])
```

Scenario	NMC			RNN-LSMC		
	RMSE	time	time*	RMSE	time	time*
1	0.884	4602.16 sec	920432 sec	0.963	801.56 sec	160322 sec
2	0.87	5343.42 sec	1068684 sec	0.863	795.17 sec	159044 sec
3	0.895	3860.51 sec	772102 sec	0.915	797.62 sec	159524 sec

- JAGS를 이용해 데이터를 샘플링하고 **시뮬레이션**을 진행했으며 NMC는 LSMC보다 RMSE가 항상 높았습니다.
- **정확도**가 높지만 복잡한 시뮬레이션으로 인해 **시간효율성**이 떨어져 이를 보완하기 위해 LSMC에 RNN을 접목해 보았습니다.
- LSMC의 빠른 시뮬레이션 속도를 사용하고 과거 데이터를 효과적으로 활용하여 미래를 예측하는 데에 특화되어 있는 **RNN** 알고리즘을 적용해서 정확도를 보완하였습니다.

고유정

T커머스 소비예측 및 편성 최적화 방안 도출

2021 빅데이터 콘테스트 (2020.05 ~ 2020.09)

#Python #R #COVID-19 #Data Analysis #Deep Learning #Machine Learning

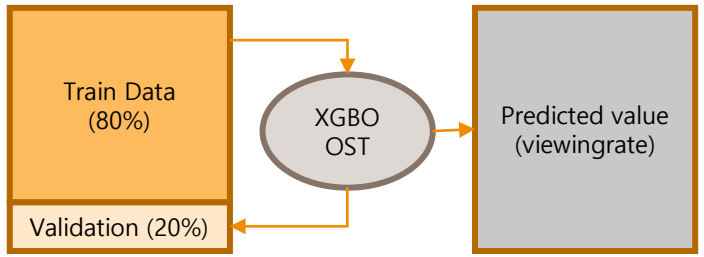
Data Preprocessing

1) 범주형 변수
새로운 상품분류와 상품코드 생성해서 데이터 재구성 및 상품 재분류

One hot encoding 사용하여 정제

2) 연속형 변수
2019년 시청률을 validation하여 2020년 시청률 예측

Min Max Scaler 사용하여 정제



Modelling

앙상블 모델 3개 이용하여 Kfold 기반 스태킹 모델 사용

MAPE가 가장 낮았던 Gradient Boosting 이용하여 훈련 진행

앙상블 모델로 예측한 값을 x값으로 넣고 스태킹 했을 때 score가 향상됨을 발견

매출 최적화 방안

요일 별 총 매출액의 중앙값을 구하고 그 값을 가진 가장 가까운 날짜를 구해 모든 시간대 고려함

코로나의 영향력을 반영한 언택트지수 'package' 생성하여 예측과 최적화에 반영함

요일별/시간별/카테고리별 시각화를 통해 매출 최적화 편성방안 제시

고유정

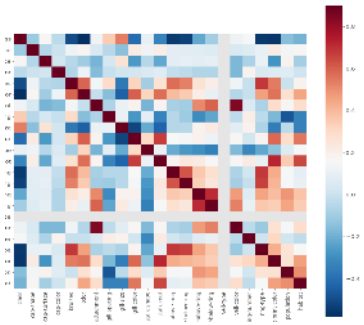
비정형 데이터분석과 최적 분류모형 구축

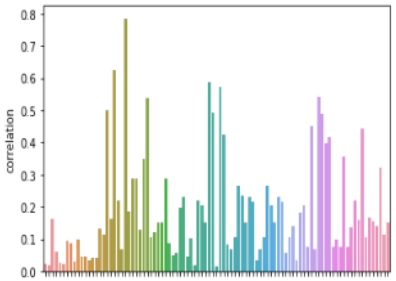
자료분석특론2 최종 프로젝트 (2020.12)

#Python #Image data #Deep Learning #Machine Learning #Classification

1) Mushroom

상관관계 분석 → Feature Importance → Accuracy 비교





Model	Accuracy
Decision Tree	0.971
Random Forest	0.974
Support Vector Machine	0.978
Logistic regression	0.879
KNN	0.985
XGboost	0.985
CNN	0.896

- Mushroom 과제에서 식용 가능한 버섯과 독성을 띄는 버섯을 구분하는 주요 특성/요인을 **분류**하고 최적 예측 모델을 탐색했습니다.
- 총 7가지 모형을 fitting시켜 **score**를 비교하고 과적합 방지를 위해 **GridSearchCV**를 사용해 하이퍼파라미터를 튜닝했습니다.
- Garbage 과제에서 분리수거 기준으로 나뉘어져 있는 비정형인 쓰레기 **이미지데이터**를 보강하고 적합시켜 최적 분류 모델을 선택했습니다.

2) Garbage

최적 모형 탐색 → 합성망이 학습한 내용 시각화

Image Data Generator, dropout, rotation range 사용한 이미지 데이터 보강

적합 후 Accuracy와 Loss를 도출하여 모형 평가 & 선택

Convolution 이용하여 순차적으로 채널 시각화

고유정

시계열 모형을 이용한 상품 수요예측 및 모형 성능비교

경제자료분석 최종 프로젝트 (2021.06)

#R #Time-series data #Data Analysis #Logistic Regression

- 자동차 수요량 예측을 위해 World Bank에서 추출한 우리나라 연별 데이터(1971 – 2019)를 사용하였습니다.

회귀분석을 통한 회귀계수 & 신뢰구간 추정

OLSE, HC, HAC, FGLSE 검정을 통한 이분산성 & 자기상관성 검정

종속변수: 연별 승용차 판매량
설명변수: 회귀분석을 통해 원유가격, 대중교통 이용량, 1인당 GDP, 면허가 있는 운전자수로 지정

→

수요탄력성 측정 시, 소비세 비율을 도구변수로 log소득을 설명변수로 설정

변수 추가 전보다 수요탄력성과 정밀도 증가

→

더빈 왓슨 검정결과 OLS에 비해 FGLSE 검정 시 자기상관 완화.

FGLSE 이용하여 신뢰구간 추정

향후 상품 수요와 예측구간 추정 & 세모형 예측력 비교

시계열 그래프 도출결과와 ADF 검정결과 통해 데이터에 확정적 추세가 있음을 확인

→

AIC order와 BIC order 구하여 ARIMA, VEC, A이 모형에 fitting하여 향후 2년 자동차 판매량 예측하고 예측구간 추정

→

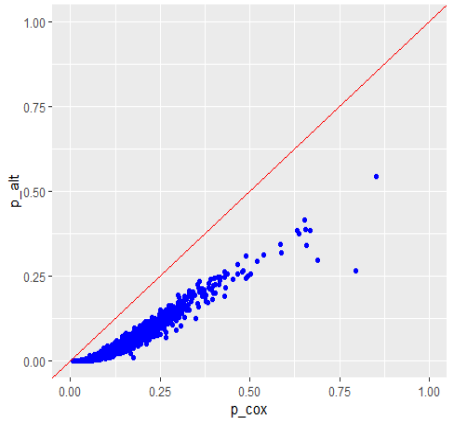
3가지 모형 비교한 결과 공적분을 고려하며 차분계열 사용해 Stepwise selection을 하고 잔차 회귀까지 정교하게 시행하여 예측한 ADL모형의 예측력이 가장 높음

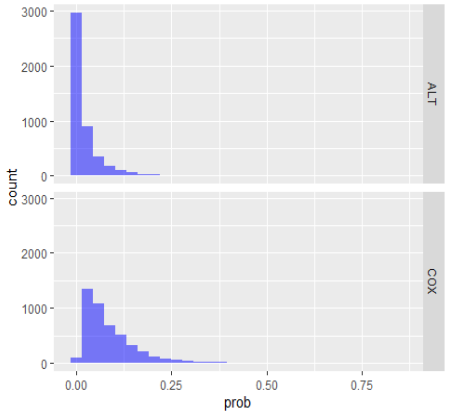
여러 종목의 데이터분석과 Rshiny 웹어플리케이션 개발

이론통계학2 팀프로젝트 (2020.09 ~ 2020.12)

#Rshiny #Data Analysis #Statistics #VBA #Power BI

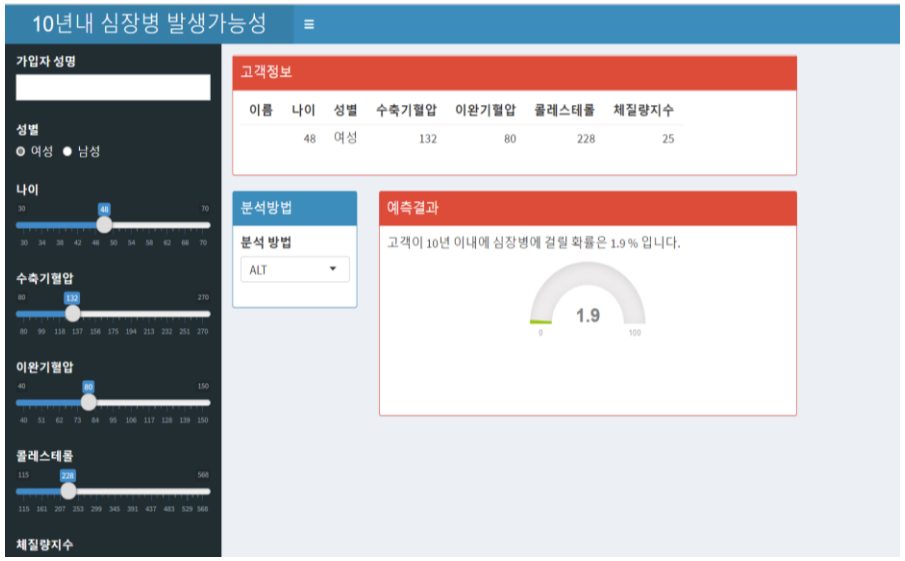
1. Cox PHM 모델 & ALT 모델을 이용한 심장병 발생확률과 AIC 비교





PHM 모델의 AIC	ALT 모델의 AIC
22738.23	31754.48

2. 구한 최적 모형을 이용하여 10년내 심장병 발생 확률을 계산하는 앱 개발



- Cox PHM/ALT 모형을 이용하여 10년내 심장병 발생가능성을 예측하고 **발생률**을 계산하는 **Rshiny application**을 개발하였습니다.
- Rshiny를 통해 **대시보드**를 구축하여 input 데이터에 따라 **개개인의** 질병 발생률을 파악하는 플랫폼을 만들었습니다.
- Cox는 ALT에 비해 데이터가 고르고 넓게 분포했고 0값이 출력되는 경우가 적었습니다. 즉, ALT의 장점은 COX PHD에 비해 과대적합을 방지하는 것임을 알 수 있었고 Cox PHM의 장점은 ALT에 비해 0값과 같은 극값의 출현 빈도가 적다는 것을 알 수 있었습니다.