다변량 데이터 분석 과제 #2: Logistic Regression & Variable Selection

리그 오브 레전드 게임 승패 예측

2018170809 오민제

Q1

게임 10분 데이터

리그 오브 레전드라는 게임은 블루와 레드 두 팀으로 나눠서 5대5로 진행되는 게임으로, 무승부가 없이 모든 판 승패가 결정되며, 평균적으로 한판에 30분 정도가 소요된다. 리그 오브 레전드에서는 승패가 결정 나지 않은, 게임 중 한 시점에서 각 팀의 유리한 정도를 정량적으로 비교하는 것은 쉽지 않고, 그 시점이 초반일수록 더욱 어렵다. 본 보고서는 게임이 초반을 넘겼을 때 인게임 데이터를 종합적으로 반영해 각 팀의 기대 승률 예측을 통해 유리한 정도를 정량적으로 비교할 수 있는 모델 구축을 목표로 한다. 이때, 초반의 기준은 게임 시작 후 10분까지로 정의했고, 다이아몬드 이상 티어에서 9,879개의 게임에 대한 40개의 변수를 바탕으로 로지스틱 회귀 모형을 사용해 기대 승률을 계산하겠다.

1. 데이터셋 (N=9,879)

Feature	Description	Property & Relationship
gameId	게임 ID	
blueWins	블루팀 승리여부	⊆ {0,1}
Blue Team (19 features)		
blueWardsPlaced	설치한 와드 수	
blueWardsDestroyed	파괴한 와드 수	≤ redWardsPlaced,
blueFirstBlood	첫 킬 여부	⊆ {0,1}, redFirstBlood와 합= 1
blueKills	처치 수	≒ redDeaths
blueDeaths	죽은 횟수	≒ redKills
blueAssists	어시스트 횟수	
blueEliteMonsters	정예 몬스터를 죽인 횟수	= (blueDragons + blueHeralds) ⊆ {0,1,2}, 0 ≤ blueEliteMonsters + redEliteMonsters ≤ 2
blueDragons	용을 죽인 횟수	⊆ {0,1}, blueDragons + redDragons ≤ 1
blueHeralds	협곡의 전령을 죽인 횟수	⊆ {0,1}, blueHeralds + redHeralds ≤ 1
blueTowersDestroyed	파괴한 포탑 수	
blueTotalGold	획득한 총 골드	
blueAvgLevel	팀 평균 레벨	
blueTotalExperience	총 경험치	
blueTotalMinionsKilled	총 처치한 미니언 수(CS)	
blueTotalJungleMinionsKilled	총 처치한 정글 몬스터 수	
blueGoldDiff	상대팀과 골드 차	= - redGoldDiff = blueTotalGold – redTotalGold

blueExperienceDiff	**상대팀과 경험치 차이**	**= -redExperienceDiff = blueTotalExperience - redTotalExperience**
blueCSPerMin	**분당 CS**	**= blueTotalMinionsKilled / 10**
blueGoldPerMin	**분당 골드**	**= blueTotalGold / 10**
Red Team (19 features)		

출처: *https://www.kaggle.com/datasets/bobbyscience/league-of-legends-diamond-ranked-games-10-min*

Fig. 1 Initial Dataset *본 데이터셋은 총 40개의 변수로 구성된다. KEY인* ***gameID****, 종속변수인* ***blueWins****를 제외한 38개의 변수는 블루팀/레드팀에 대해 각각 19개씩 서로 대칭되는 구조이기 때문에 위 테이블에서는 그 절반인 블루팀에 대한 정보만 표기하였다; 즉 모든 블루팀 변수에 대해서 위 테이블에 미 표기된 1대1 대응되는 레드팀 변수가 항상 존재한다(****blueKills ⇔ redKills****). 또한 Property & Relationship에는 해당 변수의 성질과 도메인 지식을 통해 직접적으로 존재하는 다른 변수들과 관계를 표시했다.*

현재 데이터셋에 다음과 같은 규칙으로 전처리를 진행한다.

(1) 서로 선형관계인 변수들을 제거: **GoldPerMin***10 = **TotalGold**, **CSPerMin***10 = **TotalCS** 이므로 각각 전자를 제거한다.

(2) 서로 동치인 변수들을 제거: **blueDeaths** = **redKills**, **redDeaths** = **blueKills** 이므로 후자 변수들만 사용한다. 또한 **blueFirstBlood** 와 **redFirstBlood**는 **blueFirstBlood** + **redFirstBlood**=1의 관계를 만족하는 이진 변수이므로 **FirstBlood** ⊆ **{0: 블루팀 선취점, 1: 레드팀 선취점}**, 변수 로 통합한다.

(3) 연속형 변수들을 모두 블루팀과 레드팀의 차이로 바꾼다. **blueExperienceDiff**와 같이 이미 존재하는 경우 해당 변수를 사용하고, 존재하지 않는 경우 새로 정의한다. 이 차이는 모두 블루팀을 기준으로 레드팀의 통계량을 뺀다.

(4) 범주형 변수들인 **blueDragons, redDragons, blueHeralds, redHeralds**는 각각 **Dragons**, **Heralds** ⊆ **{0: 두 팀 다 획득 실패 1: 블루팀 획득, 2: 레드팀 획득}**로 통합한다

2. 수정된 데이터셋 (N=9,879)

Feature	Description	Data Type
gameId	**게임 ID, KEY**	
blueWins	**블루팀의 승리여부**	**명목형, BOOLEAN**
FirstBlood	**첫 킬을 한 팀, {0: 블루, 1: 레드}**	**명목형, BOOLEAN**
WardsPlacedDiff	**(블루 팀이 설치한 와드 수) – (레드 팀이 설치한 와드 수)**	**이산형**
WardsDestroyedDiff	**(블루팀이 파괴한 와드 수) – (레드 팀이 파괴한 와드 수)**	**이산형**
KillsDiff	**(블루 팀 킬 수) – (레드 팀 킬 수)**	**이산형**
AssistsDiff	**(블루 팀 어시스트 수) – (레드 팀 어시스트 수)**	**이산형**
Dragons	**용을 죽인 팀, {0: 없음, 1: 블루팀, -1: 레드팀}**	**숫자형**
Heralds	**협곡의 전령을 죽인 팀, {0: 없음, 1: 블루팀, -1: 레드팀}**	**숫자형**
TowersDestroyedDiff	**(블루 팀이 파괴한 포탑 수) – (블루 팀이 파괴한 포탑 수)**	**이산형**
GoldDiff	**(블루 팀이 획득한 총 골드) – (레드 팀이 획득한 총 골드)**	**연속형**
AvgLevelDiff	**(블루 팀의 평균 레벨) – (레드 팀의 평균 레벨)**	**이산형**
ExperienceDiff	**(블루 팀 총 경험치) – (레드 팀 총 경험치)**	**연속형**
CSdiff	**(블루 팀 총 CS) – (레드 팀 총 CS)**	**연속형**
JungleMinionsKilledDiff	**(블루 팀 총 정글 CS) – (레드 팀 총 정글 CS)**	**연속형**

Fig. 2 Adjusted Dataset *KEY인* ***gameID****, 종속변수인* ***blueWins****와 13개의 설명변수로 구성된다*

Q2

분석 전 예측

본 보고서에서 분석에 사용하는 설명변수는 **2. 수정된 데이터셋**에서 **gameID**와 **blueWins** 제외한 26가지 변수들이고 종속변수는 같은 데이터셋의 **blueWins**이다. 데이터 분석에 앞서 높은 상관관계가 예상되는 변수는 **푸른색**으로, 필요 없을 것으로 예상되는 변수는 **붉은색**으로 표기하였다. 일반적으로 리그 오브 레전드에서 유리함을 비교할 때 가장 중요한 지표는 골드와 경험치 두가지이므로, 이를 중점적으로 해석해보겠다.

Feature	Type	Importance	Reasons
gameId	KEY	-	-
blueWins	종속변수	-	-
FirstBlood	설명변수	Med	첫 킬을 올린 것은 사기진작 측면에서 긍정적이기 때문에 조금이나마 영향이 있을 것이다.
WardsPlacedDiff		Med	와드를 많이 박는 행위는 게임의 승패에 긍정적인 영향을 줄 가능성이 있으나, 플레이어의들의 플레이스타일에 영향을 받기도 함으로 적당한 연관성이 예상된다.
WardsDestroyedDiff		Low	상대의 WardsPlaced가 높을수록 높을 가능성이 높음을 고려한다면, 유의하자 않을 가능성이 높다.
KillsDiff		High	적을 죽이면 일정시간 행동불능이 되므로, 그동안 충분히 게임을 유리하게 할 수 있을 것이다.
AssistsDiff		Med	높은 어시스트는 팀이 그만큼 유기적으로 움직였다는 지표일 뿐이다. 각각 라인에서 솔로킬이 많이 나와서 이기는 경우도 있기 때문에 적당한 연관성이 예상된다.
Dragons		Med	10분까지 등장하는 용은 1마리로, 첫번째 용 처치는 전략적으로 유리한 팀이 포기하는 경우도 있기 때문에, 전령에 비해서 직접적으로 승패에 큰 영향을 주지 않을 것이다.
Heralds		High	전령획득은 큰 골드 이득으로 연관되기 때문에, 게임을 유리하게 풀어가기 쉬워진다.
TowersDestroyedDiff		Med	포탑을 많이 파괴한 것은 승리에 도움이 되기는 하지만, 더욱 중요한 것은 어느 포탑을 부쉈는지 이므로 적당한 연관성이 예상된다.
GoldDiff		High	골드는 가장 중요한 두가지 지표 중 하나다.
AvgLevelDiff		High	경험치가 높을수록 레벨이 높아진다. 게임의 유리함을 직접적으로 보여주는 지표일 것이다.
ExperienceDiff		High	경험치는 가장 중요한 두가지 지표 중 하나다.
CSdiff		Med	CS를 많이 챙겼다는 것은 하나의 유리함의 지표이다. 다만, 플레이스타일의 영향을 받는 부분이 존재하기 때문에, 적당한 연관성이 예상된다.
JungleMinionsKilledDiff		Low	처치한 정글 몬스터 수는 그 팀의 5명의 플레이어 중 한명인 정글러의 플레이스타일과 보다 연관이 있는 지표로, 상관이 없을 것이다.

Fig. 3 Importance ***LOW:*** *필요 없음,* ***Med:*** *필요함,* ***HIGH:*** *필요하며 높은 상관관계가 예상됨.*

Q3

데이터의 성질

(1) 범주형

Feature	Ratio
FirstBlood	**0: 블루팀 선취점(49.52%), 1: 레드팀 선취점(50.48%)**
Dragons	**0: 드래곤 생존(22.49%), 1: 블루팀 드래곤 획득 (36.2%), -1: 레드팀 드래곤 획득 (41.31%)**
Heralds	**0: 전령 생존 (65.2%), 1: 블루팀 전령 획득 (18.8%), -1: 레드팀 전령 획득 (16.0%)**

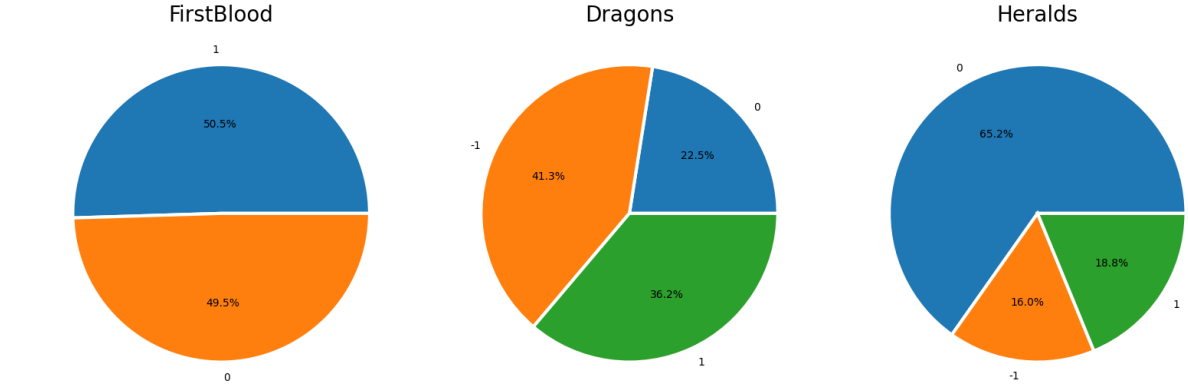
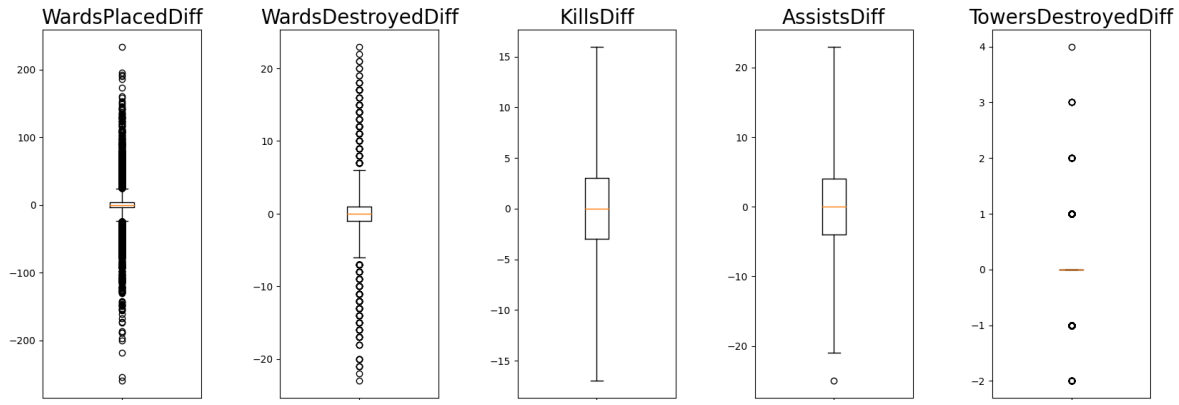


Fig. 4 명목형 변수들에 대한 설명, 파이차트를 통한 시각화

(2) 비 범주형

Feature	Mean	Sd	(Min, Max)	Skewness	Kurtosis
WardsPlacedDiff	**-0.08**	**25.96**	**(-260,234)**	**13.46**	**-0.27**
WardsDestroyedDiff	**0.10**	**2.85**	**(-23,23)**	**11.97**	**0.00**
KillsDiff	**0.05**	**4.20**	**(-17,16)**	**0.10**	**0.02**
AssistsDiff	**-0.02**	**5.77**	**(-25,23)**	**0.33**	**-0.02**
TowersDestroyedDiff	**0.01**	**0.32**	**(-2,4)**	**18.51**	**0.79**
GoldDiff	**14.41**	**2453.23**	**(-10830,11467)**	**0.30**	**0.03**
AvgLevelDiff	**-0.01**	**0.48**	**(-2.6,2.4)**	**0.58**	**0.01**
ExperienceDiff	**-33.62**	**1920.27**	**(-9333,8348)**	**0.36**	**0.02**
CSdiff	**-0.65**	**30.94**	**(-120,127)**	**0.21**	**0.02**
JungleMinionsKilledDiff	**-0.80**	**14.27**	**(-72,64)**	**0.45**	**-0.11**



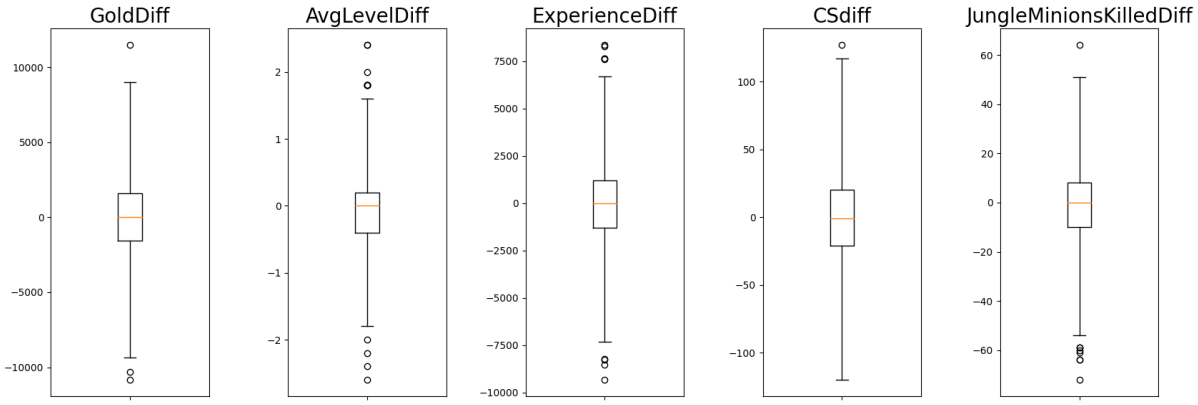


*Fig. 5 비 명목형 변수들에 대한 단변량 통계, 박스 그래프 (IQR * 2.5 기준)*

Curran et al. *1996)에 따르면 skewness와 kurtosis가 각각 (-7,7), (-2,2) 범위 내에 있을 때 정규성을 가정할 수 있다 한다. Fig 5. 에서는 WardsPlacedDiff, WardsDestroyedDiff, TowersDestroyedDiff를 제외한 변수들이 해당 조건을 만족한다. 추가적인 정규성 검정을 위해 해당 변수들에 한정해 Shapiro-Wilks 검정을 진행하고, N>5000이므로 신뢰성을 위해 QQ-plot과 Kolmogorov-Smirnov 검정을 진행한다.

Feature	SW-test		KS-test	
	Statistic	p-value	Statistic	p-value
KillsDiff	0.9973	0.0000	0.4462	0.0000
AssistsDiff	0.9973	0.0000	0.4462	0.0000
GoldDiff	0.9973	0.0000	0.4462	0.0000
AvgLevelDiff	0.9973	0.0000	0.4462	0.0000
ExperienceDiff	0.9973	0.0000	0.4462	0.0000
CSdiff	0.9973	0.0000	0.4462	0.0000
JungleMinionsKilledDiff	0.9973	0.0000	0.4462	0.0000

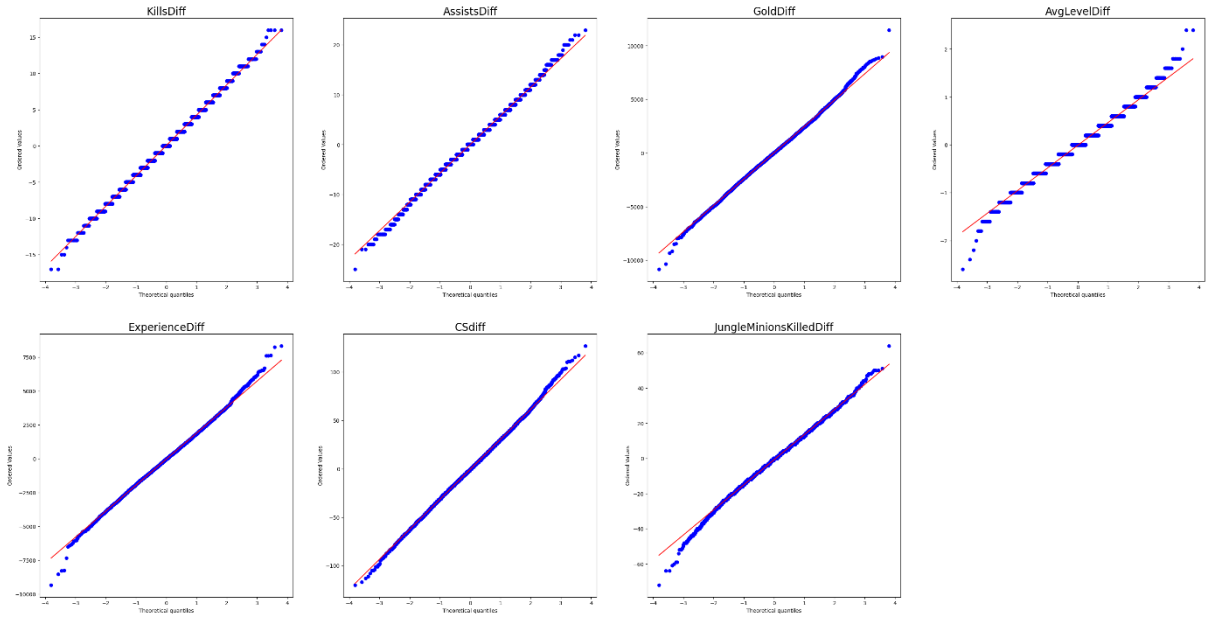


Fig. 6 가능한 변수들에 대한Shapiro Wilks test, Kolmogorov-Smirnov test, QQ-plot 결과

SW-test와 KS-test가 모두 N>5000인 경우에 대해 신뢰성이 떨어짐을 고려한다면, p-value<0.05인 경우에도 정규성을 가정할 수 있다. 이산형 변수인 **KillsDiff**, **AssistsDiff**, **AvgLevelDiff** 변수를 제외한다면, QQ-plot을 통해서 **GoldDiff**, **ExperienceDiff**, **CSdiff**, **JungleMinionsKilledDiff**에 대해서는 정규성을 가정해볼 수 있다.

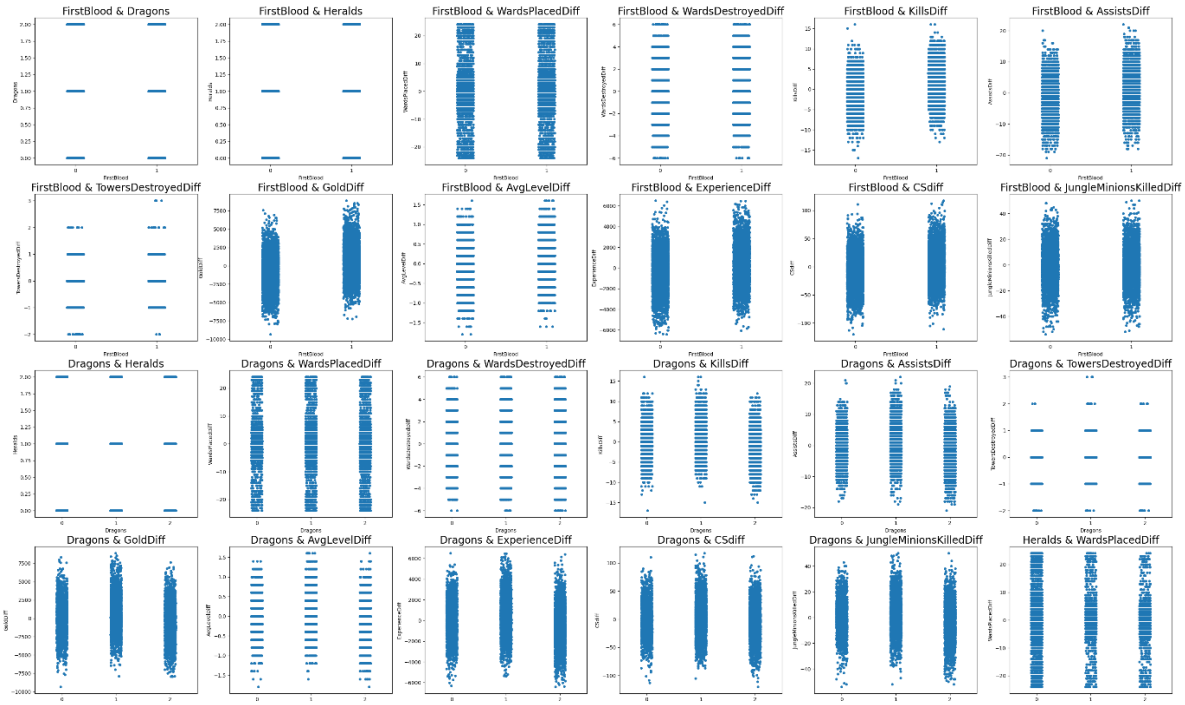
Q4

이상치 제거

리그 오브 레전드에서 게임이 기울기 시작하면 기하급수적으로 그 차이가 벌어지기 때문에 본 데이터셋에서는 Boxplot에 있어서 IQR*1.5보다 높은 2.5를 사용하는 것이 더 적절하다고 판단했다. **TowersDestroyedDiff**는이산형 변수로 대부분의 값이 0이기에 Fig.5 와 같은 형태를 띤다. 만약 0 아닌 값을 모두 제거하면 해당 변수를 사용하는 것이 무의미해질 것이다. 따라서 **TowersDestroyedDiff**를 제외한 9개 변수에 대해 (Q1-IQR*2.5, Q3+IQR*2.5) 범위를 밖을 이상치로 정의하고 제외한다. 이때 **WardsPlacedDiff**는 많은 이상치가 존재한다. 일반적으로 리그 오브 레전드에서 한 팀에서 10분동안 박을 수 있는 와드의 개수는 한정적이며, 10-20개를 넘을 수 없다. 이상치들은 이를 초과하는 100, 200등의 불가능해 보이는 값이기에, 비록 이상치가 많지만 모두 제거하는 것이 정당하다. 최종적으로 이상치를 모두 제거한 데이터는 (N=8,016)으로 1863개의 이상치가 제거되었음을 알 수 있다.

Q5

변수간 상관관계



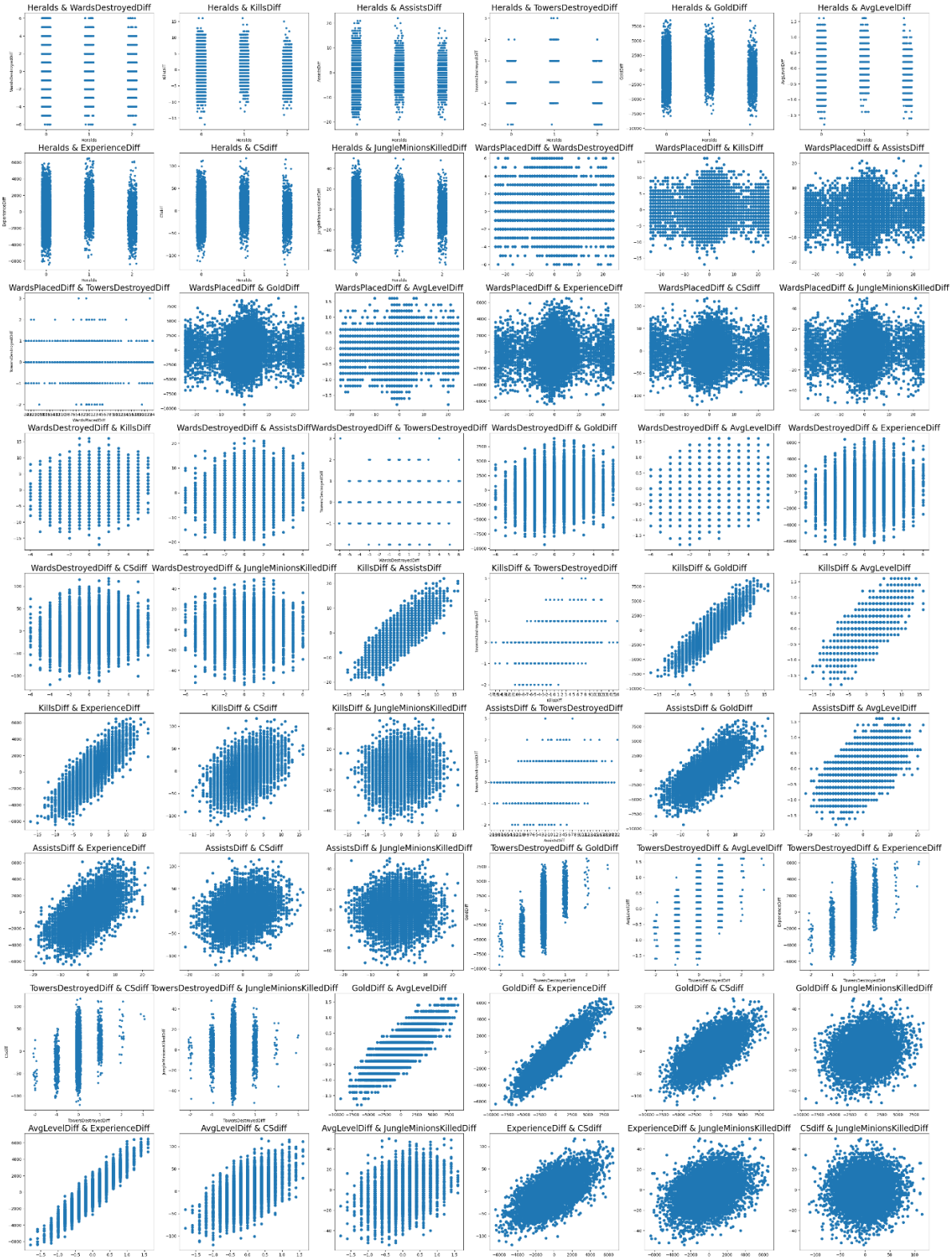


Fig. 7 가능한 모든 변수 조합에 대한 scatterplot

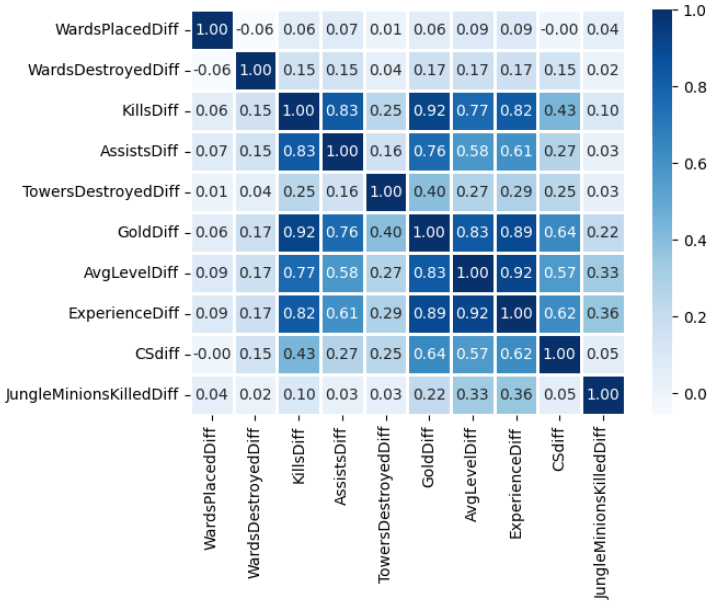


Fig. 8 명목형이 아닌 변수들에 대한 corrplot

일반적으로 상관계수가 0.8이 넘을 때 강한 상관관계를 가진다고 한다. **Fig 7**의 산점도와, **Fig 8**의 상관계수를 바탕으로 통해서, 다음과 같은 변수 쌍들이 서로 높은 상관관계가 존재한다고 할 수 있다.

(KillsDiff – AssistsDiff)

(KillsDiff – GoldDiff)

(KillsDiff – ExperienceDiff)

(GoldDiff – ExperienceDiff)

(GoldDiff – AvgLevelDiff)

(AvgLevelDiff – ExperienceDiff)

따라서 다음과 같은 변수 조합이 생기며 이 중 **GoldDiff**를 대표로 사용하겠다:

{KillsDiff, AssistsDiff, GoldDiff, ExperienceDiff, AvgLevelDiff}

** 이 외에는 강한 상관관계를 가지는 변수 조합이 존재하지 않는다.*

Q6

Logistic Regression 모델 구축

데이터셋을 7:3으로 분할하고 로지스틱 회귀분석 모델을 구축하였다

Feature	p-value	Feature	p-value
constant	0.7254	AssistsDiff	0.1458
FirstBlood	***0.0419***	***TowersDestroyedDiff***	***0.0286***
Dragons	0.4015	***GoldDiff***	***0.0000***
Heralds	**0.5130**	**AvgLevelDiff**	**0.7245**
WardsPlacedDiff	0.6276	***ExperienceDiff***	***0.0000***
WardsDestroyedDiff	**0.8530**	***CSDiff***	***0.0283***
KillsDiff	***0.0000***	**JungleMinionsKilledDiff**	0.8541

Fig. 9 로지스틱 회귀분석 결과 변수들에 대한 p-value, 사전 예측에서 중요하다고 생각한 변수들을 색으로 표시하였다. 또한 유의수준 0.05를 만족하는 변수들은 이탤릭체로 표시하였다. ***LOW:*** *필요 없음,* ***Med:*** *필요함,* ***HIGH:*** *필요하며 높은 상관관계가 예상됨*

학습 데이터의 {0,1} 비율이 50:50에 가깝기 때문에 cutoff를 0.5로 설정하였고, 로지스틱 회귀분석을 진행한 후 모든 변수의 유의수준은 Fig 9와 같다. 유의수준 0.05에서 유의한 변수는 **FirstBlood**, **KillsDiff**, **TowersDestroyedDiff**, **GoldDiff**, **ExperienceDiff**, **CSDiff** 총 6개이다. **GoldDiff**, **ExperienceDiff**, **KillsDiff**는 예측과 똑같이 중요한 변수들이며, **Heralds**와 **AvgLevelDiff**는 예측과 다르게 중요하지 않은 변수들이다. 다만 **AvgLevelDiff**의 경우 **ExperienceDiff**와 매우 높은 상관관계를 보이는, 독립이 아닌 변수이기 때문에 유의하지 않게 나왔을 가능성을 고려해야만 한다. 사전 예측과 다르게 **FirstBlood** **TowersDestroyedDiff**, **CSDiff** 들은 유의한 변수들이다. **TowersDestroyedDiff**, **CSDiff, FirstBlood** 모두 중요하다고 말한 요소인 골드에 직접적으로 큰 이득을 주는 변수들이기 때문에, 유의하더라도 상식 선에서는 말이 된다. 다만 현재 독립이 아닌 변수들이 존재하기 때문에 보다 높은 정확도를 위해 변수 선택을 진행하겠다.

Confusion Matrix		Predicted	
		1	0
Actual	1	864	323
	0	333	885

TPR(Recall)	Precision	TNR	ACC	BCR	F1
0.727	0.733	0.728	0.727	0.727	0.730

Fig. 10 로지스틱 회귀분석 결과 변수들에 대한 confusion matrix, SimpleAccuracy,BalancedCorrectionRate,F1-Measure

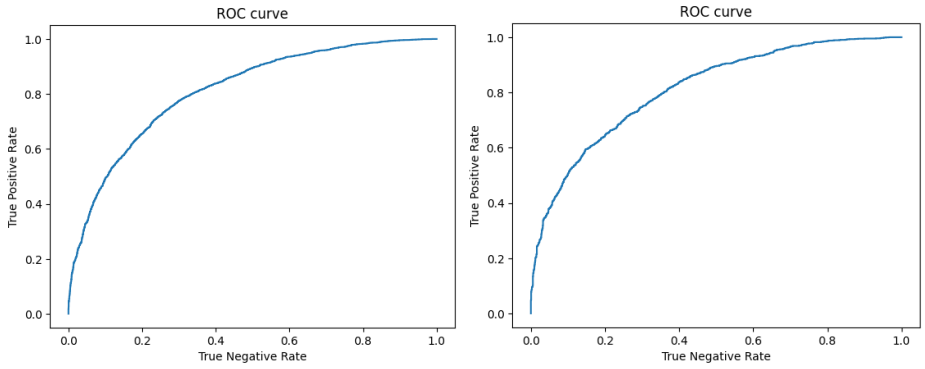


Fig. 11 로지스틱 회귀분석 모델의 ROC curve, 학습 데이터(좌, AUROC =0.81153), 검증 데이터 (우, AUROC = 0.81151)

회귀모델의 ACC(Accuracy), BCR(Balanced Correction Rate), F1(F1-measure)이 모두 유사하게 0.73 정도를 기록한다. 학습 데이터의 {0,1} 비율이 50.6:49.4로 거의 일치하기 때문에, Accuracy가 F1-measure과 Balanced Correction Rate과 그리 큰 차이가 없이 나타나는 것을 알 수 있다. 또한 현재 예제에서는 블루팀 승/레드팀 승리 중 더 중요하게 여겨지는 사건이 없기 때문에, 그냥 Accuracy를 사용해서 로지스틱 회귀분석 모델의 성능을 가늠해도 충분하다고 결론지을 수 있다.

ROC 커브의 경우, 학습 데이터와 검증 데이터에서 비슷한 양을 띠며, AUROC 또한 각각 0.81153과 0.81151로 소수점 아래 4자리까지 동일한 것을 확인할 수 있다. 이는 모델이 학습 데이터의 예시를 다 외워버린(과적합이 된) 것이 아니라, 적절하게 학습을 진행한 모델이기 때문이라고 해석이 가능하다.

Q7

휴리스틱한 변수 선택 & Logistic Regression 모델 구축

Feature	p-value	Feature	p-value
constant	0.6812	*Heralds*	*0.0023*
FirstBlood	0.6470	*TowersDestroyedDiff*	*0.0005*
WardsPlacedDiff	0.4117	*GoldDiff*	*0.0000*
WardsDestroyedDiff	0.6585	CSDiff	0.5402
Dragons	*0.0000*	*JungleMinionsKilledDiff*	*0.0085*

Fig. 12 로지스틱 회귀분석 결과 변수들에 대한 p-value, 사전 예측에서 중요하다고 생각한 변수들을 색으로 표시하였다. 또한 유의수준 0.05를 만족하는 변수들은 이탤릭체로 표시하였다. LOW: 필요 없음, Med: 필요함, HIGH: 필요하며 높은 상관관계가 예상됨

높은 상관관계를 보인 변수들을 제거하고 진행한 로지스틱 회귀분석 결과이다. 유의수준(0.05)를 만족하는 변수들은 **Dragons, Heralds, TowersDestroyedDiff, GoldDiff, JungleMinionsKilledDiff** 5가지이다. 변수제거 과정에서 제거된 **KillsDiff, ExperienceDiff**를 제외하면 **FirstBlood**, **CSDiff**가 제외되었고, **JungleMinionsKilledDiff**, **Dragons**, **Heralds**가 추가되었다. 총 유의한 변수 개수는 감소했지만, 변수 중 비율은 (선택 후:0.56>기존: 0.46)으로 증가했음을 볼 수 있다. Heralds는 예측과 동일하게 유의한 변수가 되었으며, 비록 사전 예측에서 JungleMinionsKilledDiff가 유의하지 않을 것으로 예상했으나, 현재 13개의 변수 모두 상식적으로 높으면 게임이 조금이나마 유리해지는 변수들이기 때문에 현재 유의변수 목록이 상식에 위배되지는 않는다.

Confusion Matrix		Predicted 1	Predicted 0
Actual	1	860	327
	0	343	875

TPR(Recall)	Precision	TNR	ACC	BCR	F1
0.718	0.728	0.725	0.721	0.721	0.723

Fig. 13 로지스틱 회귀분석 결과 변수들에 대한 confusion matrix, SimpleAccuracy,BalancedCorrectionRate,F1-Measure

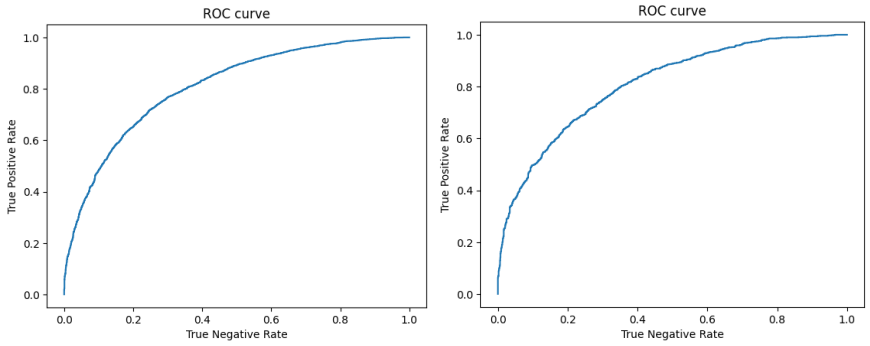


Fig. 14 로지스틱 회귀분석 모델의 ROC curve, 학습 데이터(좌, AUROC =0.80828), 검증 데이터 (우, AUROC = 0.80920)

회귀분석 결과 여전히 ACC, BCR, F1이 모두 비슷한 값을 가짐을 알 수 있다. 이는 학습 데이터의 {0,1} 비율 자체가 달라지지 않았기 때문이다. 다만 기존과 비교해서 세 값 모두 소폭 감소했음을 알 수

있으며, AUROC 또한 검증 데이터 기준으로 0.80920으로 약 0.3% 감소한 것을 볼 수 있다. 다만, 이는 유의미한 차이라 볼기는 어렵고, 머신러닝의 기본 원칙에서 비슷한 성능을 낼 수 있다면, 더 단순한 모델을 사용하는 것이 권장되기 때문에, Q6의 모델보다 좋은 모델이라고 생각할 수 있다.

ROC 커브의 경우, Q6의 모델과 동일하게, 학습 데이터와 검증 데이터에 대해 유사한 모양을 보이며, AUROC또한 0.80828, 0.80920으로 유사하다. 따라서 현재 모델이 과적합된 모델이 아닌, 건전하게 패턴을 학습한 모델이라고 해석할 수 있다.

Q8 & 9

정량적인 변수 선택 & Logistic Regression 모델 구축

정량적인 변수 선택을 위해 Forward Selection, Backward Elimination, Stepwise Selection, Genetic Algorithm 네가지 방법론들을 활용하겠다. Accuracy, BCR, F1-score, AUROC 및 소요시간, 변수 감소율을 활용해 각 방법론들의 장단을 비교하겠다.

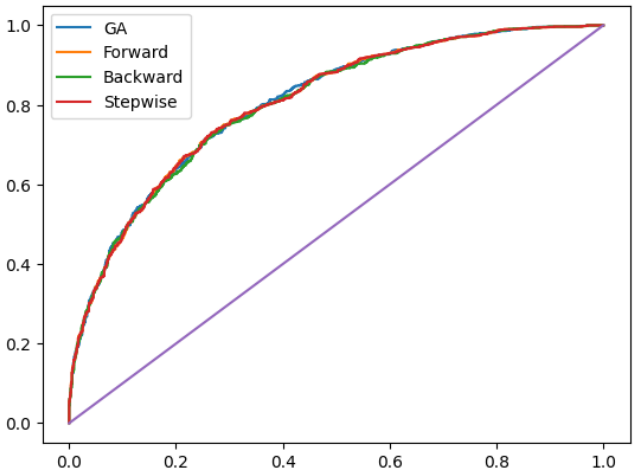


Fig. 15 4가지 정량적 방법론들의 로지스틱 회귀분석 모델의 training set ROC curve

Methods	TPR	Precision	TNR	ACC	BCR	F1	Time(sec)	AUROC	Redu %
Full	0.727	0.733	0.728	0.727 (*2)	0.727 (1)	0.730 (2)	-	0.812 (1)	0.0% (7)
Reduced	0.718	0.728	0.725	0.721 (*5)	0.721 (*3)	0.723 (5)	-	0.809 (2)	30.8% (3)
Forward	0.719	0.733	0.717	0.725 (4)	0.718 (6)	0.726 (4)	7.08 (3)	0.805 (4)	61.5% (1)
Backward	0.723	0.735	0.720	0.721 (*5)	0.721 (*3)	0.728 (3)	6.29 (1)	0.803 (6)	7.7% (6)
Stepwise	0.728	0.737	0.725	0.731 (1)	0.726 (2)	0.733 (1)	7.02 (2)	0.805 (*4)	53.8% (2)
GA	0.720	0.736	0.719	0.727 (*2)	0.719 (5)	0.728 (3)	171.55 (4)	0.806 (3)	15.4% (4)

*Fig. 16 Q7의 변수선택, Full Feature, 4가지 정량적 방법론들의 성능 지표들 비교. Forward, backward, stepwise는 f1-scoring을 이용했고, forward, backward의 변수 선택 시 최소 개선치 = $1x10^{-4}$, GA는 population = 20, mutation rate = 0.1, crossover rate = 0.3, generation = 5, fitness function = AUROC로 설정하고 진행하였다. 괄호 안 수치는 좋은 순서이다, 공동일 경우 *표시*

Method	Feature Lists
Forward	**KillsDiff, Dragons, Heralds, GoldDiff, ExperienceDiff**
Backward	**FirstBlood, WardsPlacedDiff, WardsDestroyedDiff, KillsDiff, AssistsDiff, Dragons, Heralds, TowersDestroyedDiff, GoldDiff, AvgLevelDiff, CSdiff, JungleMinionsKilledDiff'**
Stepwise	**KillsDiff, Dragons, Heralds, GoldDiff, ExperienceDiff, JungleMinionsKilledDiff**
GA	**FirstBlood, Dragons, WardsPlacedDiff, WardsDestroyedDiff, KillsDiff, AssistsDiff, TowersDestroyedDiff, GoldDiff, ExperienceDiff, CSdiff, JungleMinionsKilledDiff**

Fig. 17 4가지 정량적 방법론들의 선택된 변수 목록

우선 균형 잡힌 데이터셋을 활용하기 때문에 4가지 모델의 Accuracy, BCR, F1-score가 각각 유사한 것을 확인이 가능하다. 통념과 유사하게 Forward, Backward 기법이 가장 적은 시간을, Stepwise는 그보다 조금은 높은 시간이 필요로 하고, GA는 많은 시간을 소요한다. 또한 모든 방법론의 성능지표가 전부 유사한 것을 확인할 수 있는데, 통념처럼 Stepwise가 Forward, Backward보다 전반적으로 조금 개선된 성능을 보여주는데 비해, GA는 그리 우수한 성능을 보여주지 못하고 있다. 다만, Forward, Backward, Stepwise는 F1-score를 기준으로 진행했고, GA는 AUROC를 기준으로 진행했기 때문에 전자 방법론들은 상대적으로 우수한 F1-score를 후자인 GA는 상대적으로 우수한 AUROC를 기록했음을 확인이 가능하다. GA는 상대적으로 낮은 variable Reduction rate을 가지고 있으며, Forward와 Stepwise는 높은 variable reduction rate을 보여준다. 다만 backward가 낮은 variable reduction rate을 보여주는 것이 특이한데, 이는 현재 데이터셋이 변수를 어떻게 선택하는지에 따라서 로지스틱 회귀분석 모델이 그리 큰 차이를 보여주지 못하기 있기 때문에 나타나는 현상이라고 해석이 가능하다. 현재 모든 정량적 변수 선택기법이 공통적으로 선택하는 변수는 **KillsDiff**, **Dragons**, **GoldDiff** 세가지이다.

모든 방법론의 성능 지표가 비슷하게 기록되기 때문에, 가장 적은 시간 복잡도와 가장 높은 변수 감소율을 보여주는 Forward selection이 현재 데이터셋에 가장 좋은 차원축소 기법이라고 결론지을 수 있다.

Q10

유전 알고리즘 Hyperparameter 조정

유전 알고리즘의 hyperparameter를 조정하면서 변수 선택 결과의 차이를 알아보도록 하겠다. Fitness Function은 기존과 동일하게 AUROC를 사용하며, 다음과 같이 hyperparameter를 조정하겠다.

- ✓ population size = {5, 20, 35},
- ✓ Mutation rate = {0.001, 0.1, 0.5}
- ✓ Cross-over rate = {0.1, 0.5, 0.7},

총 27가지 조합에 대한 변수 선택 결과와, 해당 변수 조합을 사용할 때의 AUROC, Accuracy를 비교하면 Fig. 18과 같다. 우선 Crossover rate은 현재 변수 선택에 아무런 영향을 주지 못하고 있고, Mutation Rate이 증가할수록 선택한 변수의 개수는 감소하고 있으며, Population의 경우 높을수록 변수를 더 많이 선택하는 것을 알 수 있다.

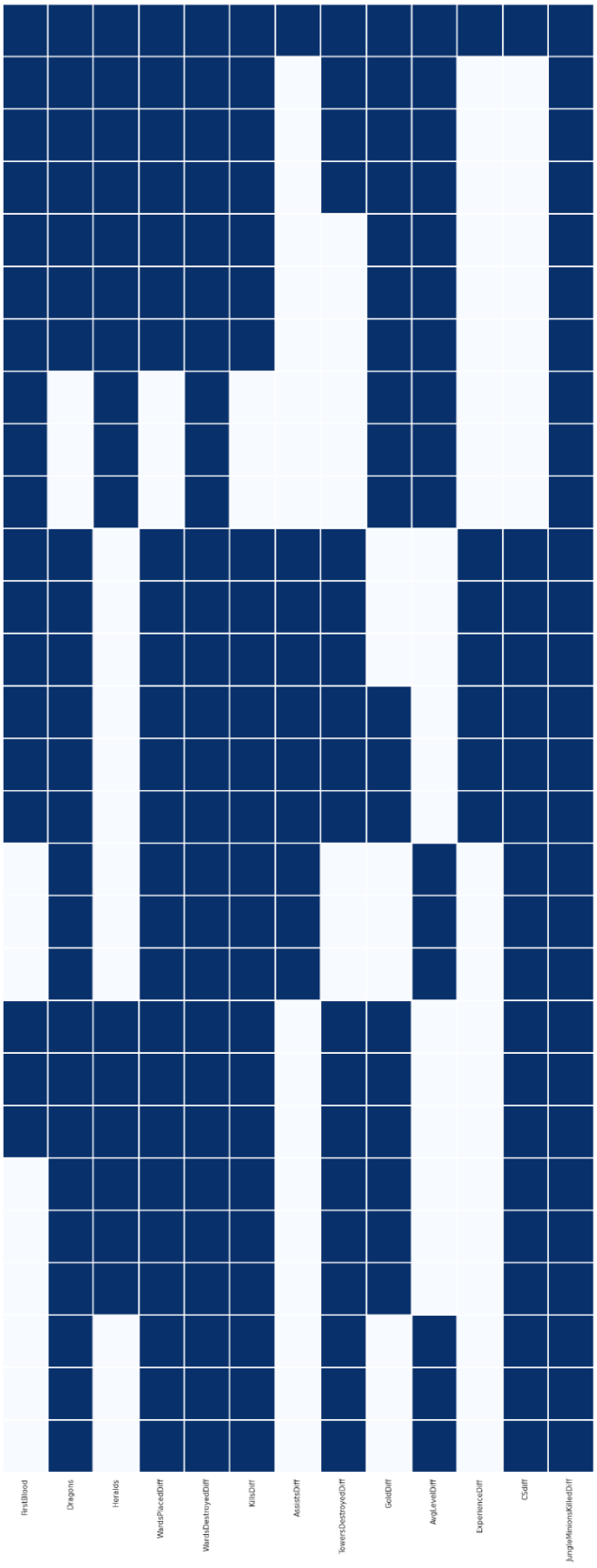
Pop	Mut	Cross	Features	Accuracy	AUROC
Full Variables (NOT GA)				0.7272	0.8115
5	0.001	0.1		0.7251	0.8113
5	0.001	0.5		0.7251	0.8113
5	0.001	0.7		0.7251	0.8113
5	0.1	0.1		0.7222	0.8107
5	0.1	0.5		0.7222	0.8107
5	0.1	0.7		0.7222	0.8107
5	0.5	0.1		0.7172	0.8043
5	0.5	0.5		0.7172	0.8043
5	0.5	0.7		0.7172	0.8043
20	0.001	0.1		0.7143	0.8059
20	0.001	0.5		0.7143	0.8059
20	0.001	0.7		0.7143	0.8059
20	0.1	0.1		0.7215	0.8121
20	0.1	0.5		0.7215	0.8121
20	0.1	0.7		0.7215	0.8121
20	0.5	0.1		0.7125	0.7983
20	0.5	0.5		0.7125	0.7983
20	0.5	0.7		0.7125	0.7983
35	0.001	0.1		0.7229	0.8099
35	0.001	0.5		0.7229	0.8099
35	0.001	0.7		0.7229	0.8099
35	0.1	0.1		0.7136	0.8040
35	0.1	0.5		0.7136	0.8040
35	0.1	0.7		0.7136	0.8040
35	0.5	0.1		0.7128	0.7983
35	0.5	0.5		0.7128	0.7983
35	0.5	0.7		0.7128	0.7983

Fig. 18 hyperparameter tuning에 따른 GA의 변수 선택, 해당 변수 조합에 대한 로지스틱 회귀분석 모델의 Accuracy와 AUROC

평가 지표들을 살펴보면 우선 population이 증가할수록 두 성능지표가 동시에 개선되고 반대로 mutation은 증가할수록 두 성능 지표가 동시에 감소되는 것을 확인할 수 있다. Hyperparameter의 변화에 따른 모델의 변화를 다음과 같이 정리할 수 있다.

	Accuracy	AUROC	Feature Reduction
Population	Increase	Increase	Decrease
Mutation	Decrease	Decrease	Increase
Crossover	None	None	None

Fig. 19 hyperparameter가 각각 증가할 때 미치는 영향

항상 적당한 수준의 hyperparameter를 정하는 것은 모델의 성능과 직결되기 때문에 중요하다. 현재 자료를 바탕으로 높은 population과 성능지표를 하락시키지 않을 정도로 적당히 높은 mutation rate인 (35, 0.1, 0.5)를 선택하는 것이 가장 타당하다고 할 수 있다. 다만, 높은 population은 그만큼 computing power를 요구하기 때문에 population을 높이기만 하는 것이 꼭 바람직하지는 않을 것이다. 모든 경우에 공통적으로 선택된 변수들은 **JungleMinionsKilledDifference, WardDestroyedDiff**, 두가지이다. 이 둘은 아이러니하게도 사전 예측에서 상관관계가 낮을 것으로 예측된 두 변수들이다. 다만 현재 mutation rate가 매우 높은 값까지 올라가기 때문에, 현재 population이 그러한 경우에는 5세대 이후로 충분히 수렴을 하지 않았을 수 있다는 점을 고려해야 한다.

BONUS

Q8, 9에서 세가지 변수 **GoldDiff, ExperienceDiff, Dragons**만이 공통적으로 사용되었다는 것을 바탕으로 이 세가지만을 활용한 모델을 마지막으로 구축해 보겠다. 리그 오브 레전드에서 플레이어의 캐릭터가 강해지는 요소는 주요변수 예측에서 언급했던 **Gold**(돈)과 **Experience**(경험치) 두가지뿐이고, **Dragon**s을 제외한 모든 변수들은 결국 일정량의 돈과 경험치를 제공한다(예를 들어 적을 처치하거나 포탑을 파괴하면 많은 양의 돈과 경험치를 제공한다). Dragon의 경우 유일하게 이 두가지와 독립적으로 캐릭터를 강화시켜주는 변수이다. 따라서 이 세가지만을 이용해서 모델링을 시도해보는 것은 유의미한 시도일 것이다.

Model	AUROC	Accuracy	BCR	F1
3 features	**0.812**	**0.728**	**0.726**	**0.724**
Best	**0.812**	**0.731**	**0.727**	**0.733**

Feature	p-value
constant	**0.7869**
Dragons	**0.0000**
GoldDiff	**0.0000**
ExperienceDiff	**0.0000**

Fig. 20 3개의 변수에 대한 로지스틱 회귀모델의 성능지표, 변수의 유의성. Best는 Q8&9의 성능지표 테이블에서 종목별 가장 우수한 값들이다.

기존 모델들과 비교했을 때 모든 성능 지표가 그리 큰 차이가 나지 않으며, p-value측면에서 모든 변수들이 통계적으로 유의한 것을 알 수 있다. 현재 데이터셋이 변수 조합에 따라 그리 큰 성능의 차이가 나지 않는 이유가 데이터에 내제된 차원이 변수 개수(=13)에 비해 매우 작은, 2-3차원 정도이기 때문에 이러한 현상이 발생한다고 결론지을 수 있다. 본 보고서에서 다룬 모든 로지스틱 회귀분석 모델들의 성능 지표는 유사하기 때문에, 마지막 모델이 변수가 가장 적어 우수하다고 결론지을 수 있다.