

Master 1 Data Analyst Année 2023-2024

Prédiction de la victoire d'une
équipe au bout de 10 minutes de
jeu : comment augmenter ses
chances de victoire ?



Gabriel Rozières

Table des matières

1.	<i>Préambule</i>	3
2.	<i>Introduction à la base de données et explications des variables</i>	4
3.	<i>Analyse descriptive</i>	5
4.	<i>Analyse inférentielle</i>	8
A.	Tests statistiques	8
B.	Modèles de régression	10
C.	Choix du modèle	13
5.	<i>Interprétation, limites du modèle et conclusion</i>	15

1. Préambule

D'ordinaire mes samedis commencent par une session jeu-vidéo avec mes amis, cependant depuis quelques semaines ils commencent plutôt par un cours de R (le langage de programmation, pas la lettre ...).

Dans le cadre de ce dit cours, nous avons eu comme projet final l'élaboration d'un projet d'analyse statistique sur la base de notre choix avec pour seule contrainte l'obligation de la traiter sous R.

Étant joueur depuis mon plus jeune âge, je me suis dit autant allier l'utile à l'agréable en travaillant sur une base de données qui m'intéresserait réellement.

Je me suis donc penché vers un de mes jeux préférés : League Of Legends.

N'étant pas spécialement un bon joueur (à peine Or après 7 ans sur le jeu ...), je me suis penché vers une problématique¹ qui me permettrait à la fois de m'améliorer et de comprendre quels facteurs sont les plus déterminants dans le but de remporter le plus de parties.

Pour ce faire j'ai réussi à dégoter une base d'environ 10000 parties de « haut niveau » (diamant), chacune de ces parties correspond à une observation. Chaque observation comporte toutes les actions stratégiques qu'un joueur peut réaliser, il en ressort une 15 aine de variables par équipe.

La base de données est issue du site Kaggle.²

¹ CF titre sur la page de garde

² Lien de la base de données : <https://www.kaggle.com/datasets/bobbyscience/league-of-legends-diamond-ranked-games-10-min>

2. Introduction à la base de données et explications des variables

Avant d'aller plus loin, voici un résumé très bref de l'objectif de ce jeu afin de comprendre le sens et l'enjeu de l'analyse : « League of Legends est un jeu de stratégie en équipe dans lequel deux équipes de cinq champions s'affrontent pour détruire la base adverse »³.

Afin de rendre l'analyse plus intuitive voici un court résumé des variables qui seront utilisées par la suite dans l'analyse :

- **Wards** (placées et/ou détruites) : il s'agit de balises qui une fois posées permettent pendant un temps donné de donner la vision de la zone à toute son équipe sans forcément s'y trouver. En les agrégeant nous obtenons le **score de vision**.
- **Kills Morts et Assists** : il s'agit dans l'ordre : du nombre de joueurs ennemis éliminés, du nombre de fois où un champion ennemi nous a éliminé et du nombre de champions ennemi auxquels on a participé à l'élimination.
- **KDA** : en combinant les 3 variables précédentes on peut obtenir un ratio qui se nomme le KDA.
- **EliteMob** : dans le jeu il y a certains monstres dits « d'élite » qui rapportent des bonus importants à toute l'équipe, ils représentent des objectifs secondaires. Ces monstres dits « élite » se nomment le « drake » et le « herald ».
- **Gold et XP** : l'or permet d'acheter de l'équipement l'XP (expérience) est nécessaire afin de monter de niveau.
- **CS et JGCS** : il s'agit de monstres, une fois éliminés ils rapportent Gold et XP ; les JGCS sont les monstres spécifiques à la jungle, ils sont plus difficiles à éliminer et rapportent donc plus de Gold et d'XP.
- Les **Tours** : elles sont les protections de la base ennemie il est nécessaire de les détruire afin de pouvoir gagner « naturellement » la partie (hors reddition de l'équipe ennemie).

Pour des soucis de vraisemblance de l'information nous privilégierons les variables des totaux plutôt que des différences (TotGold plutôt que GoldDiff) car bien que plus précises quant à l'état de la partie il s'agit d'informations dont le joueur ne peut pas avoir connaissance.

Nous nous baserons donc sur des informations plus accessibles au joueur.

La base de données étant composée de variables des 2 équipes (bleue et rouge) et donnant les mêmes informations (le nombre de Morts de l'équipe bleue est le nombre de Kills de l'équipe rouge, etc.) nous nous concentrerons donc sur une seule équipe.

³ Citation tirée du site officiel du jeu, trouvable au lien minimisé suivant : <https://urlz.fr/oh8W>

3. Analyse descriptive

Afin d'avoir une vision plus agrégée de certaines variables nous utiliserons dans la partie descriptive de l'analyse de cette base de données des valeurs comme le KDA⁴ le nombre d'« ÉliteMob » dans le but de comprendre les tendances globales liées à ces variables.

Commençons par générer la matrice des corrélations afin de voir quelles variables semblent être liées et quel est le sens de ces éventuelles relations.

Score_vision -	0.01	0.02	-0.01	0.04	0.02	0.02	0.01	0.01	0.03	0.04	0.04	-0.02	0.01	0.01	0.02	1
Objectifs -	0.24	0.22	-0.21	0.17	0.94	0.7	0.64	0.49	0.33	0.22	0.25	0.14	0.18	0.21	1	0.02
KDA -	0.33	0.41	-0.54	0.41	0.19	0.19	0.07	0.13	0.46	0.38	0.42	0.23	0.07	1	0.21	0.01
JGCS -	0.13	-0.11	-0.23	-0.13	0.2	0.16	0.12	0.01	0.09	0.37	0.41	0.17	1	0.07	0.18	0.01
TotCs -	0.22	-0.03	-0.47	-0.06	0.12	0.09	0.08	0.09	0.28	0.51	0.57	1	0.17	0.23	0.14	-0.02
TotXP -	0.4	0.47	-0.46	0.3	0.23	0.18	0.15	0.14	0.68	0.9	1	0.57	0.41	0.42	0.25	0.04
NivMoy -	0.36	0.43	-0.41	0.29	0.2	0.16	0.13	0.12	0.62	1	0.9	0.51	0.37	0.38	0.22	0.04
TotGold -	0.42	0.89	-0.16	0.75	0.24	0.19	0.15	0.35	1	0.62	0.68	0.28	0.09	0.46	0.33	0.03
Tour -	0.12	0.18	-0.07	0.12	0.17	0.04	0.22	1	0.35	0.12	0.14	0.09	0.01	0.13	0.49	0.01
Herald -	0.09	0.08	-0.1	0.03	0.64	0.02	1	0.22	0.15	0.13	0.15	0.08	0.12	0.07	0.64	0.01
Drake -	0.21	0.17	-0.19	0.17	0.78	1	0.02	0.04	0.19	0.16	0.18	0.09	0.16	0.19	0.7	0.02
EliteMob -	0.22	0.18	-0.2	0.15	1	0.78	0.64	0.17	0.24	0.2	0.23	0.12	0.2	0.19	0.94	0.02
Assists -	0.28	0.81	-0.03	1	0.15	0.17	0.03	0.12	0.75	0.29	0.3	-0.06	-0.13	0.41	0.17	0.04
Morts -	-0.34	0	1	-0.03	-0.2	-0.19	-0.1	-0.07	-0.16	-0.41	-0.46	-0.47	-0.23	-0.54	-0.21	-0.01
Kills -	0.34	1	0	0.81	0.18	0.17	0.08	0.18	0.89	0.43	0.47	-0.03	-0.11	0.41	0.22	0.02
Victoire -	1	0.34	-0.34	0.28	0.22	0.21	0.09	0.12	0.42	0.36	0.4	0.22	0.13	0.33	0.24	0.01
	Victoire	Kills	Morts	Assists	EliteMob	Drake	Herald	Tour	TotGold	NivMoy	TotXP	TotCs	JGCS	KDA	Objectifs	Score_vision

Grâce à ce tableau nous pouvons identifier la corrélation entre les variables, nous allons décrire plus amplement certaines de ces variables ci-dessous afin de mieux rendre compte de la distribution de ces dernières (sans les quartiles que nous traiterons par la suite) :

Summary Stats					
	écart-type	médiane	min	max	range
EliteMob	0.63	0.0	0	2	2
Tour	0.24	0.0	0	4	4
TotGold	1535.45	16398.0	10730	23701	12971
TotXP	1200.52	17951.0	10098	22224	12126
TotCs	21.86	218.0	90	283	193
KDA	2.58	1.5	0	49	49
Score_vision	18.22	19.0	5	254	249

⁴ Combinaison du nombre de Kills, de Morts et d'Assists selon la formule suivante : $KDA = (2 * Kills + Assists) / 2 * Morts$

Continuons en regardant les moyennes conditionnelles des variables les plus « importantes » (selon mon expérience en tant que joueur) mais aussi selon la matrice des corrélations ci-dessus.

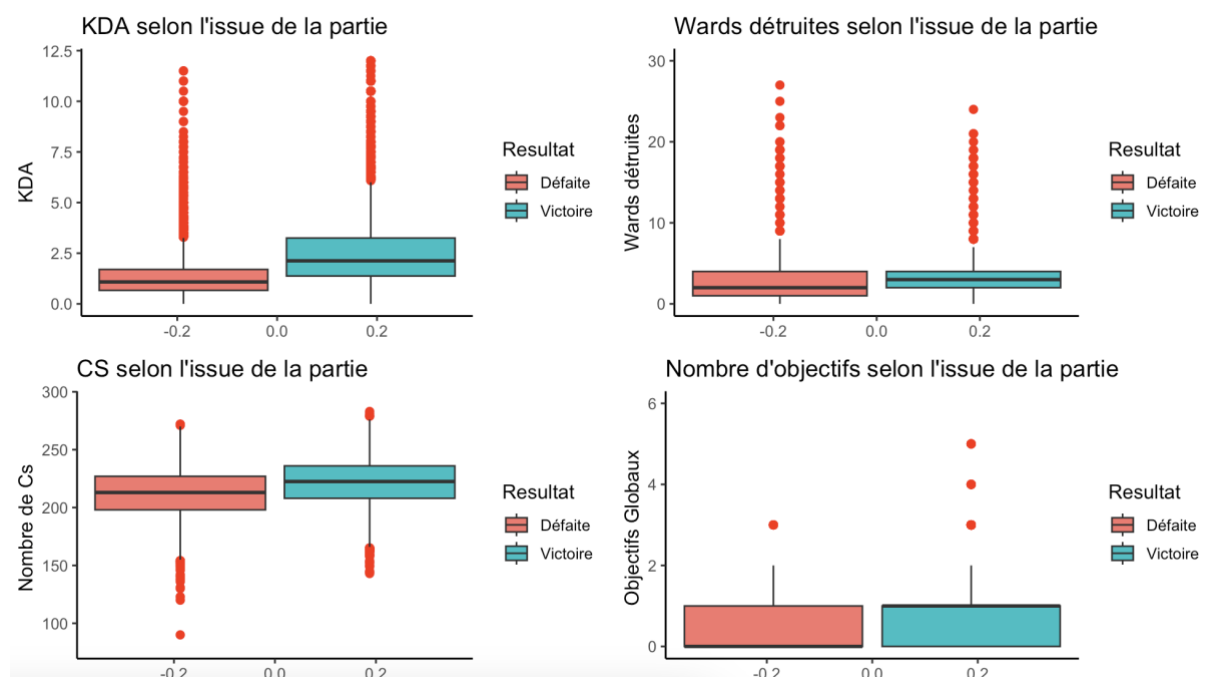
Moyennes selon l'issue de la partie						
Resultat	Nb_Gold	Nb_XP	Nb_CS	KDA	Nb_Elite_Mob	Nb_Tours
Défaite	15,864.1	17,453.5	211.8	1.4	0.4	0.0
Victoire	17,145.3	18,404.6	221.6	3.1	0.7	0.1
n: 2						

Comme nous pouvons le voir, les différences entre ces deux groupes bien que présentes ne sont pas si flagrantes qu'on aurait pu l'imaginer.

On peut aussi remarquer que le nombre moyen de tour est très faible aussi bien dans le cas d'une victoire qu'en cas de défaite, en nous penchant sur la base nous voyons qu'il n'y a en réalité que 464 parties sur les 9879 que comporte la base où une tour est tombée dans les 10 premières minutes de jeu.

Cette information nous sera très utile plus tard lors de la création de nos modèles.

En tenant compte des précédents tableaux, de la matrice de corrélation ci-dessus ainsi que de mon expérience de joueur, approfondissons encore un peu plus notre vision de certaines variables grâce à des diagrammes en boîte.



Ces graphiques nous permettent de voir la distribution des variables dont il est question et ainsi de comparer les distributions de ces variables selon l'issue de la partie. Il en ressort, comme nous pouvions nous y attendre, qu'à l'unanimité le groupe vainqueur des parties a une distribution globalement plus élevée pour ces variables.

Penchons-nous davantage sur la distribution du nombre de wards placées (plus le nombre de wards placées est élevé meilleure est notre vision de jeu ce qui augmente l'efficacité de nos stratégies).



Il semble en ressortir que malgré le fait que globalement il y ait un peu plus de wards posées pour les équipes victorieuses, mais les 2 distributions semblent très proches l'une de l'autre comme nous pouvons le voir sur le deuxième graphique.

Une question nous vient donc tout de suite en tête : le nombre de wards placées influe-t-il réellement sur l'issue de la partie ?

4. Analyse inférentielle

A. Tests statistiques

En se basant uniquement sur les statistiques descriptives ci-dessus ne pourrait-on pas supposer que les wards posées n'influent pas sur la victoire ?

Et bien posons notre test de Student : H_0 : les wards posées n'influencent pas le résultat
 H_1 : les wards posées influencent le résultat

Voici nos résultats :

Les wards placées influent-elles sur le résultat ?	
T-Test_Result	
T_stat	0.008642238
P_value	0.9931048
Intervalle_confiance	(-0.708 ; 0.714)

En regardant la P-value qui est extrêmement élevés, on ne peut donc vraisemblablement pas rejeter H_0 et donc conclure que les wards placées n'influencent pas la victoire : résultat très étonnant !!

On pourrait interpréter le résultat comme le suivant : en diamant, le nombre de wards posées dans les 10 premières minutes ne semble pas avoir d'impact sur le résultat de la partie. Cela pourrait par exemple s'expliquer par le fait qu'à ce niveau de jeu la vision ne représente pas encore un facteur influençant l'issue de la partie.

On pourrait également en profiter pour tester une légende urbaine selon laquelle on aurait plus de chance de victoire selon si on se situe d'un côté ou d'un autre de la carte⁵ :

Le côté influe-t-il sur le résultat ?	
T-Test_Result	
T_stat	-0.1911507
P_value	0.8484114
Intervalle_confiance	(0.489 ; 0.509)

On regarde encore la p-value, et, comme on aurait pu s'en douter, on ne peut pas rejeter H_0 (le côté de jeu n'influe pas sur l'issue de la partie) là non plus.

⁵ : beaucoup de joueurs (peu importe leur niveau) admettent cela.

Notre côté de la carte ne semble pas influencer nos chances de victoire. Cette légende urbaine n'est en tout cas pas vraie pour les parties en diamant.

Afin de savoir quelles variables ont un impact sur la victoire nous allons généraliser ce test aux autres variables ce qui nous donne le tableau suivant :

Quelles variables influencent la victoire ?				
	p_value	t_statistic.t	seuil_rejet	Rejet_H0
Herald	0.0000000	9.2182763	2.576327	H0* rejeté
Drake	0.0000000	21.7422290	2.576327	H0* rejeté
TotCs	0.0000000	22.9422069	2.576327	H0* rejeté
TotXP	0.0000000	42.8815006	2.576327	H0* rejeté
TotGold	0.0000000	45.6130487	2.576327	H0* rejeté
WardsDetruites	0.0000108	4.4017927	2.576327	H0* rejeté
JGCS	0.0000000	13.1768314	2.576327	H0* rejeté
Tour	0.0000000	11.5493050	2.576327	H0* rejeté
WardsPlacees	0.9931048	0.0086422	2.576327	H0* non-rejeté
Assists	0.0000000	28.6057394	2.576327	H0* rejeté
Morts	0.0000000	-35.8496814	2.576327	H0* rejeté
Kills	0.0000000	35.6052451	2.576327	H0* rejeté

^a *H0 = la variable n'influe pas sur la victoire

Dans ce tableau nous pouvons voir la valeur de notre statistique de test et la comparer au seuil de rejet, si notre statistique est plus élevée que le seuil, on peut en conclure que la variable testée, dans notre échantillon, semble influencer le résultat de la partie. Sinon, nous pouvons encore une fois nous fier à la p-value.

Comme nous pouvons le voir, sur notre base de données, seulement les wards placées ne semblent pas influencer sur l'issue de la partie.

B. Modèles de régression

Ici nous avons deux objectifs : identifier les facteurs clé pour quantifier leur importance et évidemment essayer de tester notre modèle en lui demandant de prédire des résultats.

Pour ce faire, la variable que nous essayons de prédire est la variable « Victoire » qui vaut 1 en cas de victoire 0 sinon, en fonction d'autres variables explicatives et cohérentes avec notre base de données et la réalité du jeu.

Nous allons dans un premier temps utiliser deux modèles de régression que sont les modèles Logit et Probit sur 2/3 de notre base de données que nous appellerons « échantillon d'entraînement ».

Afin de tester nos 2 modèles ainsi entraînés, nous utiliserons notre 1/3 restant (échantillon de test) que les modèles n'ont jamais « vu » pour comparer les prédictions issues de nos deux modèles avec les issues réelles des parties, ceci dans un but de choisir le model qui, dans notre cas, sera le plus approprié.

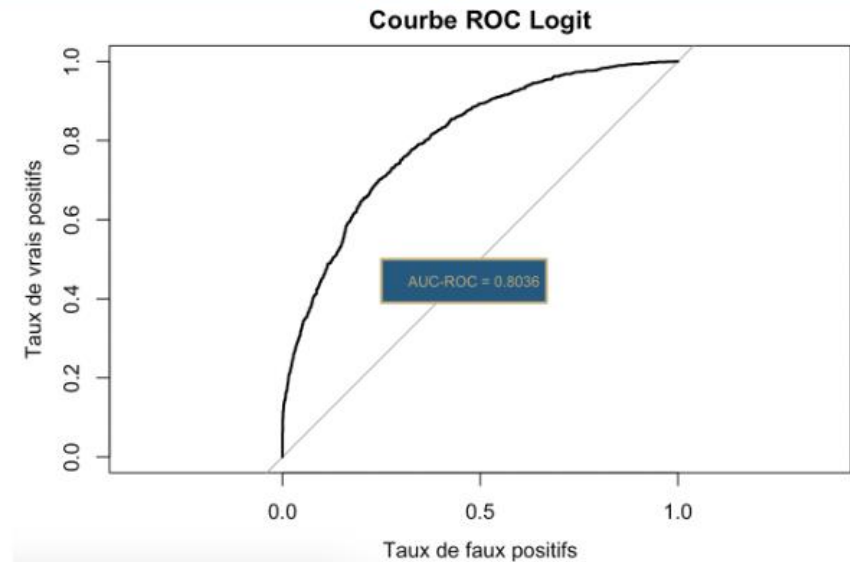
Finalement nous choisirons un seul des deux modèles afin d'en interpréter les coefficients dans le but de pouvoir en tirer des recommandations.

Modèle de régression Logistique

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 \text{Drake} + \beta_2 \text{Herald} + \beta_3 \text{TotGold} + \beta_4 \text{TotXP} + \beta_5 \text{Kill} + \beta_6 \text{Morts} + \beta_7 \text{Assists} + u$$

Dependent variable:	
Victoire	
Drake	0.297*** (0.036)
Herald	0.006 (0.045)
TotGold	0.0003*** (0.00003)
TotXP	0.0001** (0.00002)
Kills	0.030** (0.015)
Morts	-0.141*** (0.007)
Assists	-0.012 (0.008)
Constant	-5.715*** (0.427)
Observations	6,586
Log Likelihood	-3,581.851
Akaike Inf. Crit.	7,179.701
Note: *p<0.1; **p<0.05; ***p<0.01	

Évaluation de la performance du modèle



Total Observations in Table: 3293

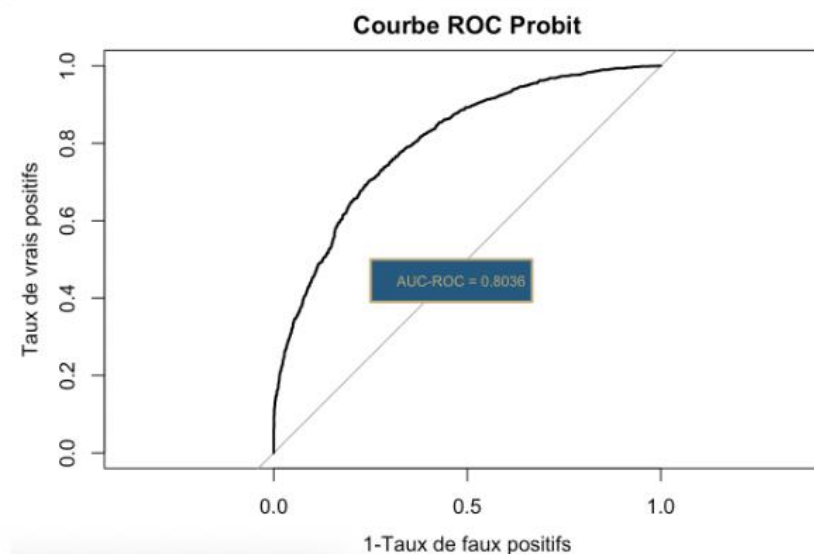
test\$ProbWin	test\$Resultat		Row Total
	Défaite	Victoire	
Loose	1179	416	1595
	0.739	0.261	0.484
Win	497	1201	1698
	0.293	0.707	0.516
Column Total	1676	1617	3293

Modèle de régression Probit

$$P(Y = 1) = \Phi(\beta_0 + \beta_1 \text{Drake} + \beta_2 \text{Herald} + \beta_3 \text{TotGold} + \beta_4 \text{TotXP} + \beta_5 \text{Kill} + \beta_6 \text{Morts} + \beta_7 \text{Assists} + u)$$

Dependent variable:	
Victoire	
Drake	0.490*** (0.061)
Herald	0.012 (0.076)
TotGold	0.001*** (0.0001)
TotXP	0.0001** (0.00004)
Kills	0.052** (0.026)
Morts	-0.239*** (0.013)
Assists	-0.021 (0.013)
Constant	-9.606*** (0.726)
Observations	6,586
Log Likelihood	-3,581.311
Akaike Inf. Crit.	7,178.621
Note: *p<0.1; **p<0.05; ***p<0.01	

Évaluation de la performance du modèle



Total Observations in Table: 3293

test\$ProbWin2	test\$Resultat		Row Total
	Défaite	Victoire	
Loose	1180	416	1596
	0.739	0.261	0.485
Win	496	1201	1697
	0.292	0.708	0.515
Column Total	1676	1617	3293

C. Choix du modèle

Que pouvons-nous conclure à partir des résultats des régressions ainsi que des méthodes d'évaluation de ces modèles ?

- Les deux modèles ont sensiblement la même aire sous la courbe ROC (arrondie à 4 chiffres après la virgule) cette courbe ROC permet de rendre compte du taux de faux positifs (TFP) et du taux de vrais positifs (TVP ou sensibilité, cas positifs correctement décelés). Cette courbe nous permet de voir à différents seuils comment le modèle arrive à discriminer les classes (victoire ou défaite dans notre cas). Une AUC-ROC (aire sous la courbe ROC) de 1 désigne un modèle qui détecterait tous les vrais positifs et n'aurait aucun faux positif. Graphiquement, la courbe ROC d'aire 1 (performance idéale) aurait une forme d'angle droit dans le coin supérieur gauche du graphique.
- La matrice de confusion est le tableau dans le rapport que vous pouvez retrouver en dessous du graphique de la courbe ROC, elle compare les valeurs prédites par le modèle (Win, Loose) avec les résultats réels (Victoire, Défaite).

D'autres métriques nous seront également utiles afin de comparer nos deux modèles, expliquons-les brièvement.

- L'accuracy de nos deux modèles, qui représente le taux de « bon classement » (le nombre de fois où le modèle prédit une victoire et que l'issue réelle est aussi victoire, idem pour les défaites). Une accuracy de 1 désignerait un modèle qui ne se trompe jamais lors de ses (et inversement si accuracy de 0) mais elle peut être trompeuse dans le cas de classes déséquilibrées. Ici, à 3 chiffres après la virgule, nos deux accuracy sont similaires.
- Le recall (taux de positifs détectés correctement par notre modèle parmi tous les positifs qui existent réellement) nous montre un très léger avantage pour le modèle Probit. Un recall de 1 indiquerait qu'aucun exemple réellement positif n'est manqué par le modèle, ce qui suggère une capacité optimale à détecter les victoires. Cependant, il est important de noter que le recall de 1 ne garantit pas la qualité globale du modèle, car un modèle prédisant uniquement des victoires, même en cas de défaites réelles, aurait également un recall de 1.
- Afin d'avoir une métrique plus précise de la performance de nos modèles combinant à la fois le recall et l'accuracy, nous utiliserons également une métrique nommée le F1 Score qui permet de fournir une évaluation globale de la capacité de nos modèles à effectuer des prédictions précises tout en évitant de manquer des exemples réellement positifs. Ici, à 3 chiffres après la virgule, nos deux F1 score sont similaires.

Voici un tableau afin de résumer les critères de décision sur le choix de notre modèle :

	Choix du modèle		
	Logit	Probit	Meilleure métrique
AUC-ROC	0.804	0.804	Égalité
Accuracy	0.723	0.723	Égalité
Recall	0.707	0.708	Modèle Probit
F1 Score	0.715	0.715	Égalité
Interprétabilité	Bonne	Moyenne	Modèle Logit

Que pouvons-nous en conclure sur nos deux modèles de manière générale ? Lequel est le meilleur ?

Premièrement on peut voir que les performances de nos modèles sont plutôt bonnes, c'est un bon signe !

Pour aller plus loin, on peut dire que globalement nos deux modèles affichent des performances plutôt similaires, nous avons donc le choix du modèle à interpréter. Nous continuerons donc notre analyse avec le modèle Logit qui a une meilleure interprétabilité ce qui nous permettra d'en tirer plus facilement des recommandations.

5. Interprétation, limites du modèle et conclusion

En reprenant les coefficients de notre régression énoncés plus tôt, dressons un tableau plus explicite afin de les expliquer plus amplement :

Coefficients du modèle Logit						
	Sign_Icon	Coeff	Inter_Conf	Effet	exp(coeff_log)	var_odd_ratio
Drake	✓✓✓	0.490467	[0.3715 ; 0.6096]	Positif	1.6330787	63.308%
Herald	✗	0.012281	[-0.136 ; 0.1607]	Positif	1.0123562	1.236%
TotGold	✓✓✓	0.000562	[4e-04 ; 7e-04]	Positif	1.0005624	0.056%
TotXP	✓✓	0.000082	[0 ; 2e-04]	Positif	1.0000819	0.008%
Kills	✓✓	0.051775	[0.0016 ; 0.102]	Positif	1.0531383	5.314%
Morts	✓✓✓	-0.239060	[-0.2642 ; -0.2142]	Négatif	0.7873677	-21.263%
Assists	✗	-0.021222	[-0.0471 ; 0.0047]	Négatif	0.9790019	-2.1%

^a ✓✓✓ = p<0.01, ✓✓ = p<0.05, ✓ = p<0.1, ✗ = p>0.1

^a intervalle de confiance à 95%

Comme nous pouvons le voir, les coefficients affectés aux variables « Herald » et « Assists » ne sont pas statistiquement significatifs c'est-à-dire que notre modèle n'estime pas significatif leurs impacts sur la variable à expliquer (ici le résultat de la partie). Ce résultat est pour le moins surprenant ! (Surtout en prenant en compte le fait que le résultat de nos tests de Student sur ces variables à la page 9 ne suggéraient pas cela).

Le coefficient de la variable « Morts » de notre modèle est quant à lui négatif (son augmentation n'est pas bon signe si notre joueur souhaite gagner !!) et statistiquement très significatif, pour aller plus loin, on peut via l'exponentielle de notre coefficient voir qu'à chaque mort de notre joueur le rapport des cotes ($\frac{p}{1-p}$) où p est la probabilité de victoire diminue de 21.263%.

Les autres coefficients de nos variables sont tous statistiquement significatifs, l'expérience (XP) et les Kills à un seuil de 95% et 99% pour les autres. Ils ont tous un effet positif sur les chances de victoire.

Interprétons rapidement un autre coefficient : les Kills ont un impact positif sur les chances de victoire de notre joueur, à chaque Kill supplémentaire, le rapport des cotes augmente de 5.314%.

En regardant ce tableau, certaines informations pourraient paraître trivial pour un amateur du jeu (essayer de mourir le moins possible, d'amasser le plus de Gold, ...) mais un coefficient pique tout particulièrement ma curiosité : celui du « Drake » je n'aurais jamais pensé qu'un Drake dans les 10 premières minutes aurait un si fort impact, il s'agit en réalité de la variable qui a le plus gros impact.

On pourrait également se demander quelles seraient les probabilités de victoire si toutes les variables explicatives de notre modèle étaient égales à 0 (si l'équipe ne joue pas).

Probabilités de victoire	
Variables	Probabilite_Victoire
Toutes les variables égales à zéro	7e-05

Comme on pouvait s'en douter, elles sont quasi nulles, en effet il me semble très peu probable qu'une équipe qui ne joue pas (sinon elle n'aurait pas toutes ses variables égales à 0) gagne, le seul cas possible serait en cas de reddition ennemie mais en partie classé ce scénario semble d'autant plus improbable.

Cela nous offre également l'opportunité de mettre à l'épreuve la robustesse de notre modèle, tout en explorant un scénario particulier qui ajoute une dimension ludique à notre analyse.

Alors que pourrais-je conseiller à un nouveau joueur qui souhaiterait maximiser son Winrate⁶ ?

Je lui conseillerais de jouer le rôle de « jungler » car il s'agit du rôle qui, donne en quelque sorte le tempo en terme de prise d'objectifs tels que le Drake, étant donné l'importance de ce dernier, je conseillerais fortement à un joueur avide de victoire de jouer ce rôle.

Prenons toutefois quelques « pincettes » sur notre analyse, il s'agit de données issues de parties classées d'un niveau Diamant, dans ces parties de joueurs plutôt confirmés (malheureusement meilleurs que moi) ne s'appliquent pas forcément de la même manière à des joueurs classés à d'autres niveaux.

De plus, cet échantillon de données n'est basé que sur des parties classées en Diamant au moment où elles ont été collectées et certains changements ont pu modérer ou accentuer l'effet de ces variables au jour d'aujourd'hui.

Ces données ne sont issues que des 10 premières minutes de jeu, un retournement de situation peut toujours apparaître au cours de la partie (j'en ai moi-même trop souvent fait les frais).

⁶ Littéralement taux de victoire

Mais la limite la plus importante à notre étude est qu'il s'agit d'un jeu d'équipe, les indicateurs sont pour la plupart agrégés au niveau de l'équipe et notre joueur n'est donc pas le seul maître de son destin.

Mais alors comment notre modèle de prédiction pourrait être appliqué en jeu ? la solution qui m'est venue en analysant le sujet serait un « add-on »⁷ qui récolterait instantanément les métriques et donnerait à notre joueur en temps réel ses probabilités de victoire lors des 1^{ères} minutes de la partie (à chaque Kill ou sbire tué il verrait sa probabilité de victoire être réévaluée).

Même si pour être honnête, bien que cette solution soit tout à fait implémentable dans le jeu techniquement en se fiant aux autres add-ons présents dans d'autres jeux, elle me paraît tout à fait antijeu (peu de gens jouerait une partie où ses chances de victoires sont très faibles) et gâcherait le plaisir et l'adrénaline des parties.

Ci-contre, un lien afin de télécharger le Script qui permet de suivre toutes les analyses statistiques réalisées : [Cliquez pour télécharger le Script R](#)

⁷ Fonctionnalité qu'on vient rajouter à un logiciel déjà existant