Rapport Mathématiques

Apollinaire Eyitayo Monteiro*
Phédeline François

Janvier 2025

Table des matières

*Apollinaire Monteiro

1 Introduction

Nous réalisons ce projet dans le cadre du cours de mathématiques avancées de la Licence 3 d'économie de l'Université Paris 1 Panthéon-Sorbonne. Un projet administré par le professeur **Marc-Arthur Diaye.**

En quoi consiste ce projet ? Ce projet subdivisé en 4 parties consiste à répondre à certaines problématiques d'ordres mathématique, statistique et économétrique.

Dans la **partie 1**, nous irons à la recherche des variables explicatives du salaire : "les features" de notre variable à expliquer qui est le salaire. En effet, ce dernier peut varier du fait qu'il s'agit du secteur, de l'entreprise ou encore du niveau d'étude. Et tous ces déterminants du salaire n'ont pas forcément les mêmes influences sur notre variable à expliquer. Par ailleurs, nous ferons d'abord une analyse descriptive de toutes les variables utilisées avant de procéder à l'estimation de notre modèle.

Dans la **partie 2**, il s'agira de construire une relation binaire entre 709 arbres, en utilisant leurs "features" de hauteur et de diamètre. Il s'agit de définir deux types de relations : la proximité I_Q, où deux arbres sont proches si leurs hauteurs et diamètres diffèrent de manière modérée selon des seuils s_1 et s_2, et l'éloignement P_Q, où un arbre est considéré plus grand ou plus large qu'un autre si sa hauteur et son diamètre dépassent ceux de l'autre de manière significative, selon les mêmes seuils. La relation Q combine ces deux notions de proximité I et d'éloignement P, et notre tâche consistera à examiner sa fermeture transitive, c'est-à-dire vérifier si, lorsque l'arbre i est lié à j, et j à k, alors i est aussi lié à k.

Dans la **partie 3**, il s'agit de résoudre un problème d'optimisation de la fonction $f(x, y, z) = x^2+y^2+z^2-2x-3y+z$ sous des contraintes définies sur un domaine D. Les contraintes imposent que $x + 2y + z \leq 4$, $x + y \geq 1$, et $x, y, z \geq 0$. L'objectif est de déterminer les valeurs de x, y, et z qui minimisent ou maximisent $f(x, y, z)$, d'abord à la main, puis de résoudre le même problème numériquement en utilisant le langage R.

En ce qui concerne la dernière partie, c'est-à-dire la **partie 4**, nous cherchons à modéliser le réseau du métro parisien en tant que graphe et à trouver un sous-graphe couvrant minimal à l'aide de l'algorithme de Kruskal. Les données proviennent du fichier *metro_distance_matrix_updated*, où $M[i, j]$ représente la distance entre deux stations. Si les stations sont sur une même ligne, la distance est donnée en nombre de stations ; sinon, elle est définie comme exp(10) pour pénaliser les changements de ligne. L'objectif est d'appliquer Kruskal pour minimiser le coût total de connexion tout en couvrant toutes les stations. Une fois ce graphe obtenu, il faudra analyser la structure du métro (par exemple, sa densité ou ses hubs principaux) et discuter du rôle stratégique de certaines stations comme La Défense et Jussieu, qui peuvent avoir une importance relative dans la connectivité globale du réseau. Cela inclut une réflexion sur leur centralité ou leur rôle dans le trafic global.

2 Fondements Méthodologiques du Projet

2.1 Partie 1 : DETERMINANTS DES SALAIRES (PAR LES MCO)

À partir de la base de données ECMOSS, nous allons pouvoir déterminer les facteurs qui expliquent notre Variable « **salaire** » en utilisant une méthode de régression par **les Moindres Carrés Ordinaires (MCO).**

Avant toute modélisation, une analyse descriptive des données est nécessaire pour explorer la distribution des variables, détecter d'éventuelles valeurs manquantes ou aberrantes, et visualiser les relations potentielles entre les variables explicatives (par exemple, l'éducation, l'expérience,

le secteur, etc.) et notre variable dépendante.

Sous certaines hypothèses de la linéarité entre nos variables explicatives et le salaire (la variable à expliquer), de la normalité des résidus (les erreurs du modèle normalement distribuées), de l'homoscédasticité (quel que soit le niveau des variables explicatives, la variance des résidus doit être constante) et de l'absence de multicolinéarité (nos variables explicatives ne doivent pas être fortement corrélées entre elles), nous pouvons passer par un modèle de régression linéaire multiple, exprimé par l'équation suivante :

$$\text{Salaire}_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \cdots + \beta_k X_{ki} + \varepsilon_i \tag{1}$$

où :

- Salaire_i est la variable dépendante (le salaire de l'individu i),
- $X_{1i}, X_{2i}, \ldots, X_{ki}$ sont les variables explicatives,
- β_0 est l'ordonnée à l'origine,
- $\beta_1, \beta_2, \ldots, \beta_k$ sont les coefficients à estimer,
- ε_i représente l'erreur aléatoire.

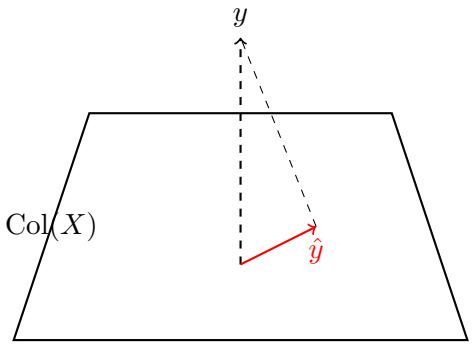
Par ailleurs, nous pouvons également faire une représentation matricielle du modèle de la manière suivante :

$$\begin{bmatrix} 1 & X_{11} & X_{21} & \cdots & X_{k1} \\ 1 & X_{12} & X_{22} & \cdots & X_{k2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & X_{1n} & X_{2n} & \cdots & X_{kn} \end{bmatrix} \cdot \begin{bmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \\ \vdots \\ \beta_k \end{bmatrix} = \begin{bmatrix} salaire_1 \\ salaire_2 \\ \vdots \\ salaire_n \end{bmatrix}$$

La méthode des MCO nous permet d'estimer les coefficients β en minimisant la somme des carrés des résidus :

$$\min_{\beta} \sum_{i=1}^{n} (\text{Salaire}_i - (\beta_0 + \beta_1 X_{1i} + \cdots + \beta_k X_{ki}))^2 \tag{2}$$

Par cette méthode, on essaiera de trouver une estimation du salaire ($\hat{y}$) qui est une combinaison linéaire des variables explicatives et qui est donc le plus proche du salaire (y). plus $\hat{y}$ sera proche de y, plus notre modèle sera performant.



L'estimateur de notre modèle est tel qu'on a :

$$\hat{\beta}_{\text{MCO}} = \left(X'X\right)^{-1} X'Y$$

Quelles sont nos variables explicatives choisies ?

pour choisir les variables explicatives, plusieurs étapes préliminaires sont nécessaires afin de garantir la pertinence de notre modèle et d'améliorer ses performances. En ce qui concerne notre projet, nous ferons un choix théorique des variables indépendantes en identifiant celles qui ont du sens d'après des études antérieures. Ensuite, nous en ferons une analyse exploratoire des données (EDA).

Dans la base (Ecomoss_2006) qui nous a permis d'identifier nos variables explicatives, nous en avons au total 132 variables (explicatives, expliquées et "inutiles"). nous distinguons les explicatives des "inutiles" compte tenu de leur pertinence dans notre modélisation. Ce sont des variables que nous considérons sans direct lien avec notre variable à expliquer comme par exemple des identifiants SIRET ou qui sont purement techniques et nécessitent plus d'informations sur leurs compositions.

Pour s'assurer du bon choix de nos variables, il est nécessaire de prendre en compte dans un premier temps la qualité de nos données. Néanmoins, si des colonnes à valeurs manquantes ne sont pas pertinentes à l'explication de notre variable dépendante, il est tout à fait logique de les exclure pour éviter toutes complexités inutiles. Par contre, il est possible qu'on ait des variables explicatives à valeurs manquantes qui soient toutefois pertinentes. Comment ? Si nous prenons l'exemple de la variable age_r (Age du salarié - variable redressée) n'ayant pas une proportion de valeurs manquantes assez conséquentes (nous parlons de $2\% < 5\%$) il serait tout à fait cohérent de quand même l'utiliser tel quel ou en supprimant tout simplement les lignes concernées de la variable car elles ont peu de chance d'introduire **un biais significatif**.

Malgré que la variable qs26_r (Diplôme le plus élevé - Valeur redressée) puisse être intéressante en ce qu'il s'agit de son effet sur le niveau du salaire d'un individu, elle n'est pas recevable. Pourquoi ? Le diplôme n'est pas renseigné par les entreprises pour environ un tiers des salariés, ce qui est une proportion conséquente des valeurs manquantes pouvant amener à un biais significatif.

Si nous prenons l'exemple des variables comme qs14_1_r (Taux de temps partiel - Valeur redressée) ou encore qs17_r (Durée du forfait en jours - Valeur redressée), ce sont des variables à valeurs manquantes qui nous importeraient peu quand il s'agira d'expliquer la variance du salaire. Mais , pas que ; il y a aussi des variables qui n'ont certes pas de valeurs manquantes

comme **moisnais** (Mois de naissance du salarié) et qui ne nous intéresse pas ; car pas pertinente pour expliquer le niveau du salaire d'un individu.

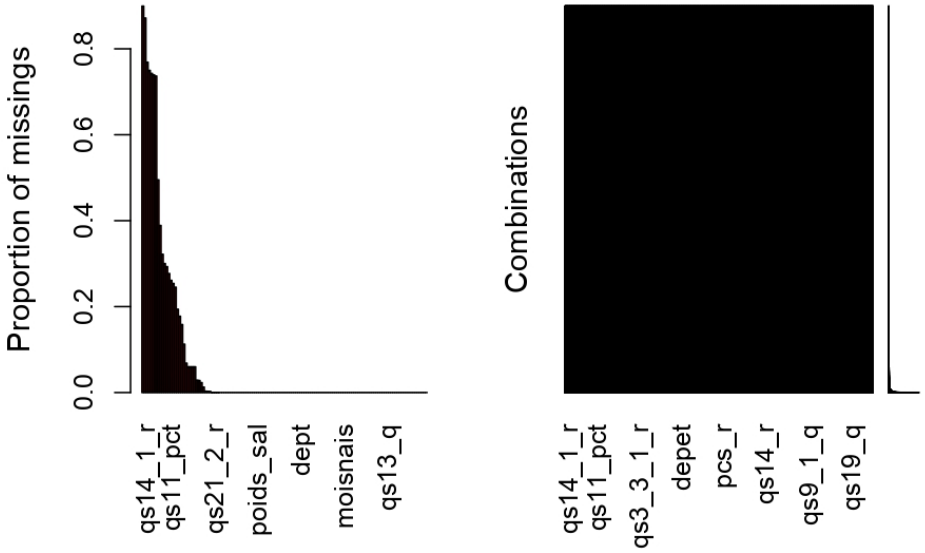


FIGURE 1 – Proportion des valeurs manquantes.

```
#Ce code nous permet de compter le nombre de valeurs manquantes
colSums(is.na(Ecmoss_2006))
```

Par ailleurs, dans l'objectif d'avoir un modèle assez performant, nous devons nécessairement nous assurer aussi que si : X_1 et X_2 sont deux variables explicatives, alors elles doivent pas être fortement corrélées. Ainsi, si nous avons par exemple des variables comme **l'ancienneté et l'âge du salarié** qui sont fortement corrélées, il est préférable de n'en garder qu'une pour éviter des redondances. Notons ce coefficient de corrélation par $\mathrm{COR}(X_1, X_2)$. On a :

$$\mathrm{COR}(X_1, X_2) = \frac{\mathrm{Cov}(X_1, X_2)}{\sigma_{X_1} \cdot \sigma_{X_2}} \tag{3}$$

où :

- $\mathrm{Cov}(X_1, X_2)$: La covariance entre les variables X_1 et X_2.
- σ_{X_1} : L'écart-type de la variable X_1.
- σ_{X_2} : L'écart-type de la variable X_2.
- $\mathrm{COR}(X_1, X_2)$: Le coefficient de corrélation, qui mesure la force et la direction de la relation linéaire entre X_1 et X_2.

Une forte corrélation se situe dans la plage $0.7 \leq |\mathrm{COR}(X_1, X_2)| < 0.9$. Et dans ce cas il serait approprié par exemple de choisir soit X_1 ou X_2.

Avant de passer à la validation des variables pertinentes, nous allons faire les choix. Si nous pouvons avoir moins de variables expliquatives et toutefois assurer une bonne estimation de

notre modèle, alors c'est ce qui est à privilégier. Notons aussi que dans notre base, nous avons des variables quantitatives et qualitatives. Nous aurons donc à faire une petite correspondance pour des variables comme par exemple Sexe_R **(la valeur 1 pour faire référence aux hommes et 2 pour faire référence aux femmes)** ou encore la variable Statut qui est binaire c'est-à-dire, dans les observations, CD pour dire que le salarié est un cadre et NC pour dire non cadre. Pour cette dernière, nous pouvons créer **une variable dummy** c'est-à-dire indicatrice : **1 pour non cadre et 0 pour cadre.**

2.1.1 Variables explicatives initialement choisies

- Sexe_r (variable qualitative) : qui représente le sexe du salarié, soit femme, soit homme
- statut (variable qualitative) : c'est le statut du salarié, soit il cadre, soit il est non cadre
- age_r (variable quantitative) : c'est l'âge du salarié
- duree (variable quantitative) : représente le nombre de jours travaillés par le salarié
- nbheur (variable quantitative) : c'est le nombre d'heures travaillées
- cs_r (variable qualitative) : c'est la catégorie socioprofessionnelle niveau 1
- cs2_r (variable qualitative) : c'est la catégorie socioprofessionnelle niveau 2
- qs24_r (variable quantitative) : Ancienneté du salarié dans l'entreprise au 1er janvier de l'année enquêtée
- qs3_3_r (variable quantitative) : c'est le total des primes et compléments de salaire

2.1.2 Test de corrélation pour réduire la multicolinéarité

Dans l'ouvrage *"Introduction to Econometrics"* de **James H. Stock et Mark W. Watson**, ils affirment que : « La présence de multicolinéarité entre les variables explicatives peut gonfler les erreurs standards, rendant les estimations instables et les inférences trompeuses. Réduire les variables fortement corrélées est donc essentiel pour garantir la robustesse et l'interprétabilité des modèles économétriques. »

Dans un premier temps, nous ferons le test pour nos variables quantitatives et dans un second temps celui de nos variables qualitatives.

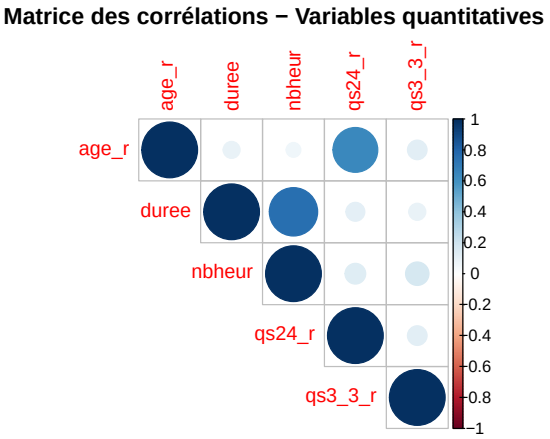


FIGURE 2 – Matrice des corrélations - Variables quantitatives

Analyse Figure 2

Les cercles indiquent l'intensité et le sens de la corrélation entre chaque paire de variables : - Plus le cercle est grand, plus la corrélation est forte

- Plus la couleur du cercle est bleu foncée, plus la corrélation est positive et tend vers 1. Si non, il serait rouge et tendrait vers -1 (absent ici).

Constat : les variables **duree et nbheur** montrent une forte corrélation positive avec un coefficient de corrélation d'une valeur de **0.752**. Nous exclurons donc l'une d'entre elles en raison de l'effet de redondance qu'elle pourrait avoir sur l'estimation de notre modèle.

Pourquoi garderons nous la variable nbheur ? parce qu'elle offre une granularité plus élevée, ce qui peut enrichir nos analyses. Deux salariés ayant travaillé 15 jours pourraient avoir des charges de travail différentes si l'un travaille 5 heures/jour et l'autre 3 heures/jour par exemple. Ainsi, la variable **nbheur** capture mieux les variations individuelles dans les horaires.

Les autres variables quantitatives ayant une corrélation faible peuvent être gardées sans souci.

Test de corrélation de nos variables qualitatives

Pourquoi avons-nous décidé de nous intéresser à la multicolinéarité de nos variables qualitatives ? Tout simplement parce que si par exemple deux de nos variables qualitatives sont fortement corrélées, cela veut dire qu'elles apportent des informations similaires, ce qui peut fausser les coefficients non observés et qu'on cherche à estimer. Il devient donc essentiel de tester la dépendance de nos variables et à quel point.

Si oui, par exemple deux de nos variables qualitatives sont fortement corrélées, la question se poserait de savoir comment allons-nous traiter la colinéarité de ces variables ? Soit on décide d'exclure l'une des variables qu'on jugera économiquement et méthodiquement moins pertinente, soit les deux variables même si fortement corrélées ont tout leur sens dans le modèle ; alors dans ce cas on pourra garder les 2 variables tout en passant par une gestion de la redondance. Par ailleurs, ce choix ne se fera pas par tâtonnement, il sera guidé par des méthodes exécutables.

Nous allons utiliser un test du chi-deux pour détecter la dépendance entre chaque paire de nos variables qualitatives. Pourquoi ? Parce qu'il évalue si deux variables catégorielles (une variable qui prend des valeurs qualitatives, représentant des catégories ou des groupes distincts) sont statistiquement indépendantes ou s'il existe une relation significative entre elles, en comparant les fréquences observées et attendues. Ce test est guidé par ce que nous appelons **une p-value**. Ce n'est simplement qu'un nombre qui indique si les résultats d'un test statistique sont dus au hasard ou non : plus elle es petite, plus il est probable que l'effet observé soit réel et non dû au hasard. Par convention, si $\boldsymbol{p < 0,05}$, on considère que le résultat est statistiquement révélateur, ce qui signifierait que nos deux variables sont dépendantes, tandis que si $\boldsymbol{p \geq 0,05}$, on conclut qu'il n'y a pas suffisamment de preuves pour rejeter l'hypothèse d'indépendance et les deux variables sont donc considérées comme indépendantes. La **figure 3** nous présente un heatmap des p-values du chi-deux.

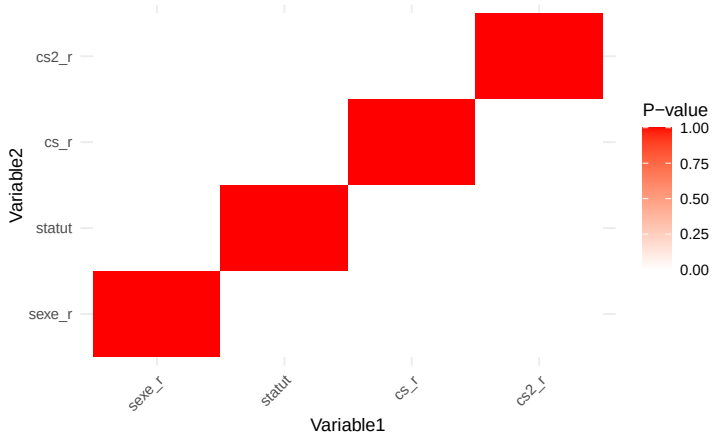


FIGURE 3 – Heatmap des p-values du chi-deux.

Analyse Figure 3

Le heatmap ci-dessus visualise les p-values des tests de dépendance entre les variables catégorielles **« sexe_r », « statut », « cs_r » et « cs2_r ».**

— Toutes les p-values (hors la diagonale) sont égales à 0, indiquant une dépendance entre chaque paire de variables.
— Les p-values sur la diagonale sont égales à 1, ce qui est attendu car chaque variable est parfaitement corrélée avec elle-même.

Malgré que nos variables qualitatives soient dépendantes, il est plus que nécessaire de savoir à quel point. Ainsi, pour connaître la force de cette dépendance, nous allons utiliser le *V de Cramér*. En effet, cela va nous permettre de savoir si cette liaison est suffisamment significative pour influencer l'estimation de notre modèle. De façon conventionnelle, dans l'interprétation du coefficient de V Cramér, on dit ceci :

– Si $V \approx 0$, cela indique une faible dépendance ou quasi-indépendance entre les variables.
– Si $V > 0.5$, cela suggère une forte dépendance.
– Si $V > 0.8$, cela peut indiquer une dépendance très forte, parfois au point de redondance **(l'une des variables apporte peu ou pas d'information supplémentaire)**.

Le graphique 4 ci-dessous nous présente un heatmap des coefficients de Cramér's V pour nos variables qualitatives.

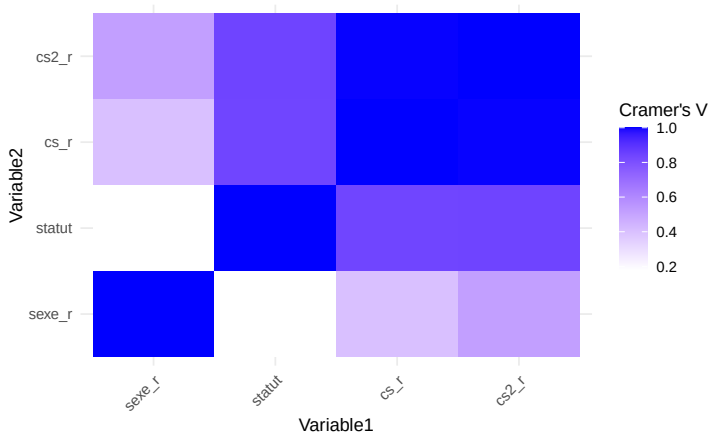


FIGURE 4 – Heatmap des coefficients de Cramer's V.

Analyse Figure 4

Matrice de Cramèr

	sexe_r	**statut**	**cs_r**	**cs2_r**
sexe_r	1.000	0.182	0.402	0.521
statut	0.182	1.000	0.838	0.842
cs_r	0.402	0.838	1.000	0.998
cs2_r	0.521	0.842	0.998	1.000

Dans la matrice ci-dessus :

- **Statut et sexe_r** ($V = 0.182$) : Ce qui indique une faible dépendance car **V est approximativement égale à 0**. Par conséquent, il est tout à fait cohérent de **garder les variables statut et sexe_r pour l'estimation de notre modèle**.

- **Statut et cs_r** ($V = 0.838$) : Ce qui montre une forte dépendance car **V est supérieur à 0,8**. À voir la définition de cs_r dans le Dico_ECMOSS, il est tout à fait logique de **supprimer cs_r**. Pourquoi ? Parce que ce dernier complexifierait inutilement notre modèle et ses valeurs prises peuvent, dans une certaine mesure, être approchées par une classification binaire identique à celle de la variable **"statut"**.

- **Statut et cs2_r** ($V = 0.842$) : Même raisonnement avec la variable **"cs2_r"**.

- **CS_R et cs2_r** ($V = 0.998$) : Ce qui confirme notre logique. Puisque les variables **CS_R et cs2_r** nous montrent une équivalence parfaite.

Conclusion :

En ce qui concerne nos variables qualitatives qui nous serviront pour expliquer le salaire, celles retenues sont : le statut et le sexe_r.

2.1.3 Variables explicatives finalement choisies

— Sexe_r (variable qualitative) : qui représente le sexe du salarié, soit femme, soit homme
— statut (variable qualitative) : c'est le statut du salarié, soit il cadre, soit il est non cadre
— age_r (variable quantitative) : c'est l'âge du salarié
— nbheur (variable quantitative) : c'est le nombre d'heures travaillées
— qs24_r (variable quantitative) : Ancienneté du salarié dans l'entreprise au 1er janvier de l'année enquêtée mesurée en mois
— qs3_3_R (variable quantitative) : c'est le total des primes et compléments de salaire

2.1.4 Analyse descriptive de nos 6 variables explicatives choisies

En quoi consisterait l'analyse descriptive ? Tout simplement à résumer et interpréter les principales caractéristiques des données associées aux variables explicatives. Cela nous permettra d'explorer leur structure avant de passer à l'estimation de notre modèle.

Analyse de nos variables quantitatives

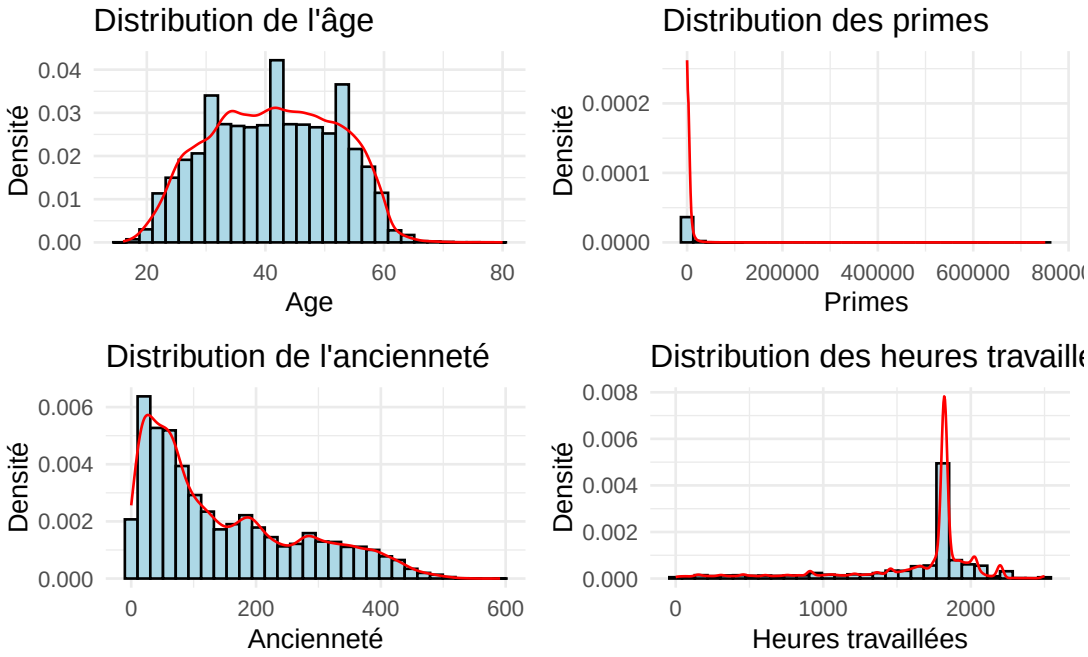


FIGURE 5 – Distribution des variables explicatives (quantitatives)

TABLE 1 – Statistiques descriptives des variables

Variable	Minimum	Maximum	Moyenne	Médiane	Écart_type
age_r	16	80	41.42381	42	10.43241
qs3_3_r	0	751037	4635.87219	2940	9399.95611
qs24_r	0	591	146.65460	103	123.94775
nbheur	0	2500	1614.06771	1820	487.47694

Distribution de l'âge

La distribution de l'âge semble relativement symétrique et suit une forme approximativement normale. Elle nous montre que l'âge moyen des travailleurs est autour de la quarantaine (un peu plus de 41 ans), ce qui reflète une population en âge actif. Nous remarquons que la moyenne et la médiane sont presque confondues. D'ailleurs, cette dernière nous permet de dire que 50% de notre échantillon a plus de 42 ans et 50% en ont moins. L'écart-type de 10,43 montre une dispersion modérée autour de la moyenne. L'âge minimum des travailleurs est de 16 ans et le maximum de 80 ans.

Distribution des primes

La variable qs3_3_r est fortement asymétrique. Notons qu'ici, notre échantillon touche en moyenne une prime de 4635.87 EURO ce qui est bien supérieure à la médiane qui est de 2940 EURO (50% de l'échantillon touche plus de 2940 EURO et 50% moins que ça). Le maximum qu'on puisse toucher en prime est extrêmement élevé, il est de 751037 EURO. L'écart-type très élevé (9399.96 EURO) montre une grande variabilité dans les données. Nous comprenons qu'un petit nombre reçoit des primes très élevées.

Distribution de l'ancienneté

La variable qs24_r est aussi asymétrique. La moyenne d'ancienneté des travailleurs étant de 146.65 mois (soit 12,22 ans), est supérieure à la médiane qui est de 103 mois (soit 8,58 ans). Ce qui indique que certaines observations ont des valeurs très élevées. L'écart-type élevé de 123,95 mois (soit 10,32 ans) reflète une forte dispersion des niveaux d'ancienneté dans l'échantillon. Le maximum d'ancienneté est de 49,25 ans.

Distribution des heures travaillées

Nous constatons que La moyenne du nombre d'heures travaillées par an répond à peu près aux normes institutionnelles soit 1614.07 heures/an. La médiane (1820 heures/an) s'en écarte légèrement, indiquant une légère asymétrie vers des heures travaillées plus faibles. L'écart-type de 487.48 heures/an montre une dispersion modérée. Notons que le maximum d'heures travaillées par un individu par an est de 2500. Cette variable est représentative d'une population active. Les valeurs qui tendent vers 0 pourraient correspondre à des travailleurs à temps partiel ou à des périodes de chômage.

Pour garantir le respect des hypothèses des moindres carrés ordinaires (MCO), la standardisation des variables comme qs3_3_r et qs24_r peut être envisageable.

Analyse de nos variables qualitatives

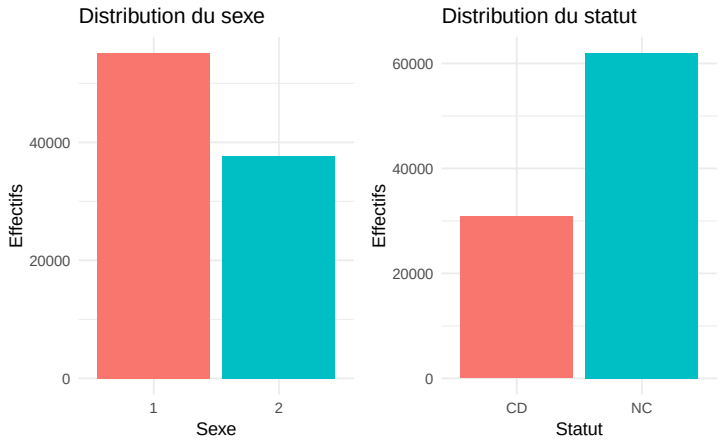


FIGURE 6 – Distribution des variables explicatives (qualitatives)

Distribution du sexe

TABLE 2 – Effectifs et pourcentages selon le sexe

Sexe	Effectifs	Pourcentage (%)
1	55116	59.382
2	37700	40.618

Nous constatons que les hommes (1) représentent 55116 individus, soit 59.38% de l'échantillon et les femmes (2) représentent 37700 individus, soit 40.62% de l'échantillon. Plus d'hommes que de femmes dans notre échantillon.

Distribution du statut

TABLE 3 – Effectifs et pourcentages selon le statut

Statut	Effectifs	Pourcentage (%)
CD	30847	33.23457
NC	61969	66.76543

Nous constatons que la catégorie "NC" (Non Cadre) représente 61969 individus, soit 66.77% de l'échantillon et que la catégorie "CD" (Cadre) représente 30847 individus, soit 33.23% de l'échantillon. Moins de salariés cadres que de salariés non cadres dans notre échantillon.

2.1.5 Tableaux de contingence

Ce qui serait encore plus intéressant à faire, c'est une analyse croisée. Pourvoir répondre à des questions plus osées ; quel pourcentage des hommes et des femmes sont cadres/non-cadres ? ou encore entre les hommes et les femmes, qui reçoivent les primes les plus élevées ?

TABLE 4 – Effectifs et pourcentages par statut et par sexe

	CD (Effectifs)	NC (Effectifs)	Somme	CD (%)	NC (%)
1	22229	32887	55116	40.33	59.67
2	8618	29082	37700	22.86	77.14
Somme	30847	61969	92816		

Avec le **tableau 4**, nous observons que parmi les 55116 hommes, 22229 sont cadres (40.33%) et 32887 sont non-cadres (59.67%). Parmi les 37700 femmes, 8618 sont cadres (22.86%) et 29082 sont non-cadres (77.14%). On peut dire qu'il y a plus d'hommes qui sont cadres que de femmes. Implicitement, les femmes sont surreprésentées parmi les non-cadres. Cela reflète des inégalités structurelles dans l'accès aux postes à responsabilité, ce qui pourrait avoir un impact sur notre variable à expliquer (le salaire). Cette disparité entre les sexes pourrait jouer un rôle clé dans l'explication des écarts de salaire. Par ailleurs, la majorité des employés, quel que soit le sexe, sont non-cadres (66.77%), ce qui est cohérent avec la structure typique de nombreux marchés du travail où les cadres représentent une minorité.

TABLE 5 – Statistiques descriptives de prime selon sexe_r

Sexe_r	Moyenne_prime	Médiane_prime	Écart-type_prime
1	5516.0	3451	11169
2	3350.0	2258	5690

Avec le **tableau 5**, nous observons que les hommes reçoivent en moyenne 5516 EUROS de primes et que les femmes en moyenne reçoivent 3350 euros de primes. Les hommes reçoivent en moyenne 2166 euros de plus que les femmes en primes. Ce qui nous laisse comprendre que les femmes ont des primes en moyenne plus faibles que les hommes. La médiane des primes pour les hommes est de 3451 euros et celle des femmes est de 2258 EUROS, ce qui est inférieur à la moyenne dans les 2 cas. L'écart-type de 11169 EUROS pour les hommes et de 5690 EUROS pour les femmes indique une dispersion des primes des hommes presque deux fois supérieure à celle des femmes. Cela suggère une plus grande hétérogénéité dans les montants des primes attribuées aux hommes. La **figure 6** présente une distribution des primes selon le sexe.

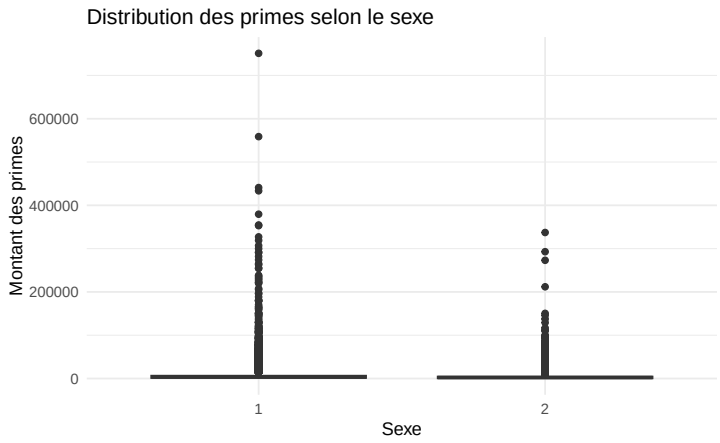


FIGURE 7 – Distribution des primes selon le sexe

2.1.6 Estimation du modèle

Nous pouvons maintenant commencer l'estimation de notre modèle. Pourquoi fait-on l'estimation ? En quoi consiste-t-elle ?

Oui, nous savons qu'il existe bel et bien des variables qui influencent le niveau du salaire d'un individu. La question que l'on se pose, de manière objective, est la suivante : Si on travaille plus d'heures, le salaire augmente, mais de combien ? Si quelqu'un a beaucoup d'ancienneté, ça joue sûrement, mais dans quelle proportion ? Et donc l'estimation nous permet de répondre à ce genre de question.Trouver des "coefficients" pour chaque facteur. L'estimateur qu'on utilisera pour y arriver dans notre cas s'appelle les Moindres Carrés Ordinaires (MCO). Il cherchera à faire en sorte que les erreurs soient les plus petites possibles entre les salaires qu'on observe dans les données et ceux qu'on prévoit avec notre modèle.

Le phénomène qu'on cherche à expliquer peut être appelé de deux manières différentes en fonction du contexte.

Nous avons **le salaire brut** qui est la rémunération totale qu'un employé reçoit pour son travail, avant toute déduction. Il est représenté dans notre base par **s_brut**. Il est entre autre composé du salaire de base qui est la rémunération pour les heures travaillées et des primes et compléments (comme les primes d'ancienneté, les primes de performance, ou encore les heures supplémentaires). Cependant, ce montant brut n'est pas ce que l'employé perçoit réellement. Pourquoi ? Parce qu'une partie de ce salaire est déduite pour financer des contributions sociales. Ce qu'il en reste, c'est son salaire réel, le salaire net perçu. Il est représenté dans notre base par **s_net**.

Cette distinction est importante à établir parce que les deux définitions n'ont pas les mêmes objectifs ni les mêmes interprétations s'agissant de l'estimation de notre modèle. En effet, le choix entre salaire net et salaire brut influence directement les objectifs et les conclusions de notre estimation. Il est donc essentiel de bien préciser quel aspect on cherche à étudier.

Compte tenu de notre objectif, il est tout à fait convenable d'utiliser la définition du salaire net. Puisque l'incidence des variables explicatives sur le salaire net (s_net) du salarié est plus directement interprétable. En outre, ce choix réfléchi nous offre une neutralité par rapport aux politiques sociales qui peuvent faire varier "le salaire brut" en fonction de réglementations ou de conventions collectives, qui ne sont pas directement influencées par les variables explicatives de notre modèle. En se concentrant sur **le salaire net**, on limite l'impact de ces variations structurelles. Quoi qu'il soit possible de l'utiliser.

Par ailleurs, les variables explicatives sont essentiellement centrées sur des caractéristiques individuelles et professionnelles du salarié, **le salaire net est donc plus représentatif.**

2.1.7 Analyses descriptives de la variable dépendante : s_net

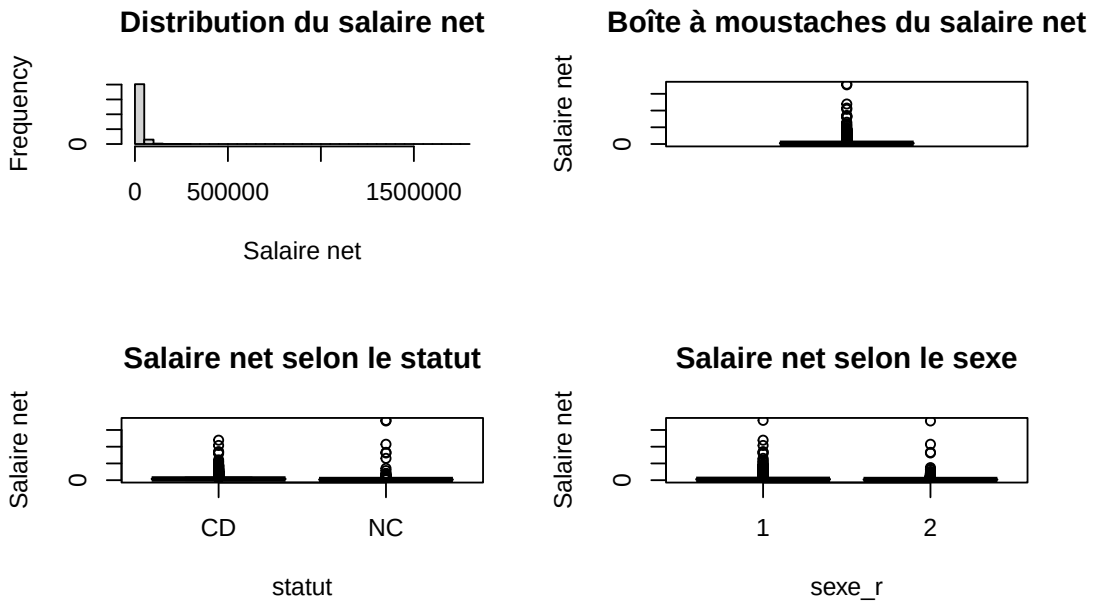


FIGURE 8 – Analyse exploratoire de la distribution du salaire net

La figure 8 qui est une représentation de l'analyse exploratoire de la distribution du salaire net, nous montre une distribution très asymétrique avec une concentration élevée de salaires faibles et quelques valeurs extrêmement élevées, ce qui confirme la présence de valeurs aberrantes. Au niveau de la boîte à moustaches, les valeurs aberrantes sont visibles au-delà des moustaches, ce qui nous indique des salaires nettement supérieurs à la majorité des données. Le statut (CD vs NC) ; montre des distributions similaires avec des valeurs aberrantes dans chaque groupe et de même pour le sexe. Nous constatons donc une très forte dispersion des salaires nets autour de leur moyenne (25474 EUROS par an), ce qui reflète une grande hétérogénéité dans les salaires au sein de l'échantillon. Cette variabilité ; dont le coefficient de variation (CV) est égale à **93,63 %** est potentiellement amplifiée par des valeurs aberrantes. Puisque le salaire maximal est déjà de 1787886 EUROS par ans. Il pourrait sembler efficace de traiter ces aberrations pour éviter d'avoir des résidus moins extrêmes ou encore un R^2 modéré.

La transformation logarithmique de nos variables très dispersées ou asymétriques pourrait nous aider dans ce sens.

```
> summary(Ecmoss_2006$s_net)
   Min.  1st Qu.   Median    Mean  3rd Qu.    Max.   NA's
      0    14678    20788    25474    30786 1787886   5559
```

2.1.8 L'équation du modèle

L'équation du modèle peut être exprimée comme suit :

$$\begin{aligned} \boldsymbol{s\_net} = \boldsymbol{\beta_0} + \boldsymbol{\beta_1} \cdot \boldsymbol{age\_r} + \boldsymbol{\beta_2} \cdot \boldsymbol{nbheur} \\ + \boldsymbol{\beta_3} \cdot \boldsymbol{qs24\_r} + \boldsymbol{\beta_4} \cdot \boldsymbol{qs3\_3\_r} \\ + \boldsymbol{\beta_5} \cdot \boldsymbol{statut} + \boldsymbol{\beta_6} \cdot \boldsymbol{sexe\_r} + \boldsymbol{\varepsilon} \end{aligned}$$

Où :

- s_net : représente le salaire net (variable dépendante, le phénomène qu'on cherche à expliquer).
- β_0 : L'intercept, la constante du modèle, salaire net moyen si toutes les variables explicatives sont nulles ; un coefficient à estimer entre autre.
- $\beta_1, \beta_2, \ldots, \beta_6$: Coefficients à estimer par les MCO, indiquant l'effet de chaque variable explicative sur s_net.
- age_r : Âge du salarié.
- $nbheur$: Nombre d'heures travaillées.
- $qs24\_r$: Ancienneté (en mois).
- $qs3\_3\_R$: Total des primes et compléments.
- $statut$ et $sexe\_r$: le sexe et le statut des individus.
- ε : Erreur aléatoire (écart entre les salaires nets observés et prédits).

Le code de notre modélisation est la suivante :

```
model_net <- lm(log(s_net + 1) ~ age_r + log(nbheur + 1) + log(qs24_r
    + 1) +
                log(qs3_3_r + 1) + statut + sexe_r,
                data = Ecmoss_2006)
```

Ce code spécifie bien un modèle de régression linéaire multiple qui permet d'expliquer notre variable dépendante qui est le salaire net (s_net). La transformation logarithmique est utilisée pour réduire la dispersion des données, atténuer l'influence des valeurs aberrantes, et gérer les valeurs nulles avec l'ajout de +1. Nos prédicteurs incluent le nombre d'heures travaillées, l'ancienneté, et le total des primes, toutes transformées logarithmiquement en raison de leur forte dispersion. Le modèle est ajusté sur les données à l'aide de la fonction lm (utilisant les MCO), qui estime les coefficients pour quantifier l'effet de chaque facteur.

2.1.9 Qu'observons nous donc ?

```
Call:
lm(formula = log(s_net + 1) ~ age_r + log(nbheur + 1) + log(qs24_r +
    1) + log(qs3_3_r + 1) + statut + sexe_r, data = Ecmoss_2006)

Residuals:
     Min       1Q   Median       3Q      Max
-9.2553  -0.2136  -0.0248   0.1847   7.2283

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)    3.969059   0.017024  233.47  < 2e-16 ***
age_r          0.006996   0.000160   43.44  < 2e-16 ***
log(nbheur + 1) 0.772006   0.002303  335.25  < 2e-16 ***
```

```
log(qs24_r + 1) 0.045976   0.001509   30.46  < 2e-16 ***
log(qs3_3_r + 1)0.041822   0.005032    8.31  < 2e-16 ***
statutNC       -0.669327   0.002886 -231.92  < 2e-16 ***
sexe_r2        -0.101586   0.002768  -36.69  < 2e-16 ***
---
Signif. codes:  0  *** 0.001 **

Residual standard error: 0.3772 on 80866 degrees of freedom
(11943 observations effacees parce que manquantes)
Multiple R-squared:  0.7639,    Adjusted R-squared:  0.7639
F-statistic: 4.36e+04 on 6 and 80866 DF,  p-value: < 2.2e-16
```

2.1.10 Analyse et Interprétation : Résultats de la régression linéaire multiple

L'objectif de notre modèle a été de percer les mécanismes qui façonnent le salaire net par la mise en lumière des facteurs clés qui influencent sa dynamique. Il cherche à voir comment des éléments comme l'âge, le nombre d'heures travaillées, l'ancienneté, les primes, le statut et le sexe peuvent expliquer pourquoi certains salaires sont plus élevés ou plus bas. Quantifier l'impact précis de ces variables, c'est ce qui nous intéresse le plus.

C'est fait ! nous avons pu atteindre notre envie d'avoir un modèle assez performant, assez fiable. En effet, il présente une excellente qualité statistique globale avec un R^2 (qui évalue dans quelle mesure le modèle capture les variations de la variable cible donnant ainsi une indication globale de la performance du modèle) ajusté de 0.7639, signifiant que **76.39% de la variance du logarithme du salaire net est expliquée par nos variables indépendantes.** La statistique F très significative (p-value < 0.0000000000000022) confirme la pertinence globale de notre modèle.

2.1.11 Compréhension de nos coefficients estimés :

1. **Variables Quantitatives :**
   - **Âge (age_r)** : Le coefficient de 0.0069696 implique qu'une année supplémentaire d'âge augmente le salaire d'environ 0.7% ($e^{0.0069696} \approx 1.007$), toutes choses égales par ailleurs.Ce qui implique que l'âge, probablement, a un effet positif mais modéré sur le salaire net.
   - **Heures travaillées (log(nbheur + 1))** : L'élasticité de 0.772 signifie qu'une augmentation de 1% des heures travaillées entraîne une augmentation de 0.772% du salaire net. Puisque la variable explicative (heures travaillées) et la variable dépendante sont en logarithme, le coefficient s'interprète directement comme une élasticité. C'est pourquoi le coefficient de 0.772 signifie qu'une augmentation de 1% des heures travaillées entraîne une augmentation de 0.772% du salaire net.
   - **Ancienneté (log(qs24_r + 1))** : Une augmentation de 1% de l'ancienneté conduit à une augmentation de 0.046% du salaire net, montrant un effet positif mais modéré. Même raisonnement que le nombre d'heure travaillé. Cette sensibilité, pourrait indiqué que la fidélité à une entreprise n'est peut-être pas suffisamment valorisée, ce qui peut justifier l'existence d'un risque potentiel de "turnover" chez les employés expérimentés.
   - **Primes (log(qs3_3_r + 1))** : Une augmentation de 1% des primes correspond à une augmentation de 0.042% du salaire net. Même raisonnement que le nombre d'heure travaillé. Cela n'est pas satisfaisant car une incidence faible ; une opportunité donc de renforcer la part variable pour stimuler la performance des salariés.
2. **Variables Qualitatives :**
   - **Statut (statutNC)** : Le coefficient -0.6693270 signifie que, toutes choses égales par ailleurs, **les non-cadres gagnent environ 48.8%** ($1 - e^{-0.6693270}$) **de moins que**

les cadres (référence dans le modèle). Ce qui est révélateur d'un écart salarial considérable entre les catégories professionnelles.Nécessité potentielle donc d'une politique de promotion interne plus dynamique.

— **Sexe (sexe_r2)** : Le coefficient -0.1015864 indique un écart salarial d'environ 9.67% ($1 - e^{-0.1015864}$) au détriment des femmes, ceteris paribus. Ce qui est révélateur d'une persistance des inégalités salariales entre les genres.

Tous nos coefficients sont hautement significatifs (p-values < 0.001), indiquant une grande fiabilité statistique des estimations. L'erreur standard résiduelle de 0.3772 est relativement faible pour un modèle logarithmique, suggérant une bonne précision des estimations.

2.1.12 Toutes les hypothèses des MCO sont-elles vérifiées ?

```
> ad.test(residuals(model_net))

    Anderson-Darling normality test

data:  residuals(model_net)
A = 948.15, p-value < 0.00000000000000022
```

Selon le test d'Anderson-Darling, notre p-valeur ($< 0.00...22$) est très inférieure à 0.05, donc on rejette l'hypothèse nulle (H0) : les résidus ne sont pas normalement distribués. Il y a donc une violation de l'hypothèse de normalité des résidus. Grâce au théorème central limite (TCL), La non-normalité des résidus n'est pas un problème majeur puisque notre échantillon est grande de taille.

```
> bptest(model_net)

    studentized Breusch-Pagan test

data:  model_net
BP = 12006, df = 6, p-value < 0.00000000000000022
```

Selon le test de Breusch-Pagan, la p-valeur ($< 0.00...22$) est très inférieure à 0.05, nous rejetons donc l'hypothèse nulle (H0) d'homoscédasticité. Il y a donc présence d'hétéroscédasticité dans notre modèle. Quoique, ce n'est pas un problème majeur car les estimateurs MCO restent non biaisés et convergents. Seule la propriété d'efficience est affectée.

	age_r	log(nbheur + 1)	log(qs24_r + 1)	log(qs3_3_r + 1)	statut	sexe_r
VIF	1.566957	1.146950	1.669854	1.201602	1.059361	1.052044

TABLE 6 – Valeurs de VIF pour chaque variable explicative

Le test de multicolinéarité préalablement fait est confirmé. Le test VIF (Variance Inflation Factor) montre que Toutes les variables ont un VIF < 5. Nous avons donc pas de problème de multicolinéarité. Les estimateurs sont donc fiables de ce point de vue.

Nos estimations étant basées sur un échantillon, pas sur toute la population. Il y a donc une incertitude qu'il faut et qu'on peut quantifier. Nous allons donc passer par **les intervalles de**

confiance pour : Quantifier l'incertitude autour des estimations des coefficients ou encore évaluer la précision des estimations.

Variable	2.5%	97.5%
(Intercept)	3.936281305	4.002930414
age_r	0.006655094	0.007284059
log(nbheur + 1)	0.767487210	0.776514040
log(qs24_r + 1)	0.043009208	0.048925255
log(qs3_3_r + 1)	0.040835888	0.042808572
statutNC	-0.674983601	-0.663670465
sexe_r2	-0.107013457	-0.096159351

TABLE 7 – Intervalle de confiance des coefficients estimés

L'analyse des intervalles de confiance à 95% révèle une précision remarquable des estimations de notre modèle salarial. L'écart salarial lié au genre est évalué avec fiabilité entre -10.15% et -9.17%. La différence entre cadres et non-cadres est encore plus marquée, se situant entre -49.06% et -48.50%. L'effet de l'âge, bien que modeste, est précisément estimé, avec une augmentation du salaire comprise entre 0.67% et 0.73% par année supplémentaire. **L'étroitesse de ces intervalles témoigne d'une grande précision de notre modèle, ce qui renforce la validité des conclusions relatives aux disparités salariales observées.**

2.1.13 Analyse des graphiques diagnostiques

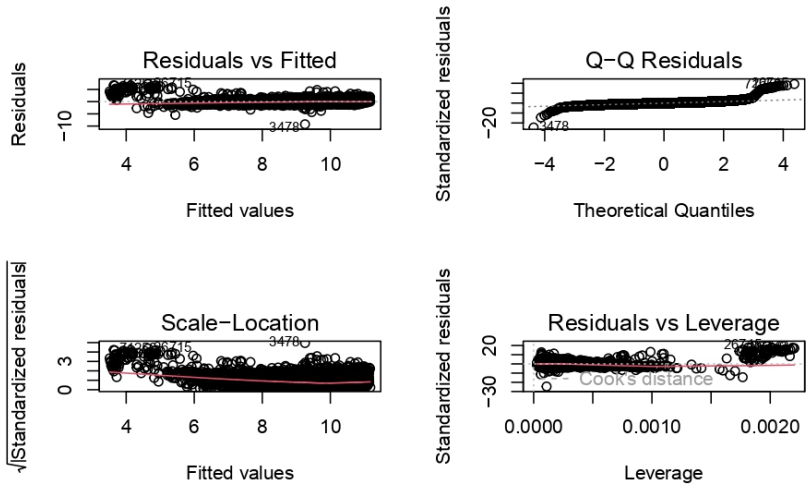


FIGURE 9 – Diagnostics visuels de la régression

Les diagnostics visuels de la régression révèlent quatre aspects majeurs de notre modèle :

- Le graphique "Residuals vs Fitted" présente un pattern non aléatoire des résidus, ce qui indique une potentielle non-linéarité dans les relations modélisées.
- L'analyse du Q-Q plot met en évidence une déviation significative aux extrémités de la distribution, confirmant la non-normalité des résidus précédemment identifiée par les tests statistiques.

- Le graphique "Scale-Location" illustre une dispersion non constante des résidus standardisés, attestant de la présence d'hétéroscédasticité dans notre modèle.
- Enfin, le "Residuals vs Leverage" identifie certains points à fort levier, sans toutefois révéler de valeurs aberrantes critiques selon la distance de Cook.

Ces observations graphiques corroborent les conclusions des tests statistiques antérieurs.

2.1.14 Et si on faisait de la prédiction ? (Facultatif)

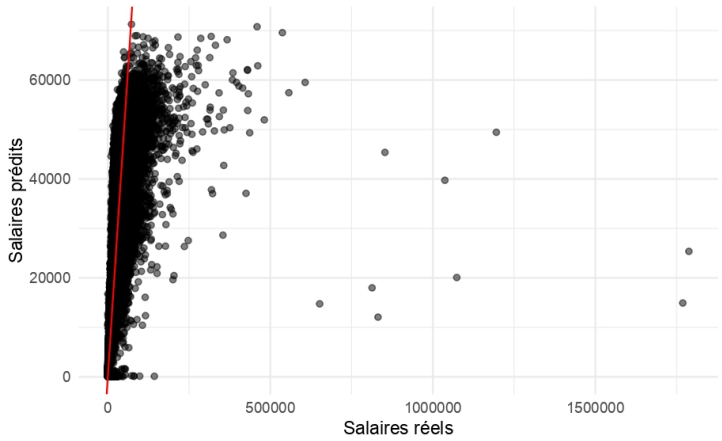


FIGURE 10 – Comparaison des salaires réels et prédits

L'évaluation de la capacité prédictive de notre modèle révèle une performance cohérente avec les objectifs d'estimation salariale. Nous avons trouvé le RMSE (Root Mean Square Error) = 19075.36 EUROS (il indique l'ampleur moyenne des écarts de prédiction) et le MAE (Mean Absolute Error) = 6871.65 EUROS (il représente l'erreur absolue moyenne des prédictions). L'intervalle de notre prédiction est tel que le salaire net est compris entre 9436.54 et 41391.98 EUROS. Notre modèle semble bien capturer les salaires faibles à moyens, mais a des difficultés à prédire correctement les salaires les plus élevés. Il a l'air de sous-estimer des salaires trop élevés. Par exemple, pour un salaire net réel de 50000 EUROS, la prédiction attendue, serait un peu plus de 20000 EUROS. Rappelons que notre modèle explique 76 % de la variance des salaires, ce qui est une performance globalement bonne. Cependant, la sous-estimation des salaires élevés (comme celui de 50000 EUROS) indique que le modèle fonctionne mieux pour les salaires faibles ou moyens, mais devient moins précis pour les valeurs extrêmes.Si les salaires élevés représentent une minorité, dans ce cas, la sous-estimation n'affectera pas significativement la qualité globale du modèle.

2.2 Partie 2 : CONSTRUCTION D'UNE RELATION BINAIRE ET FERMETURE TRANSITIVE

À première vue, établir **une relation binaire** entre des arbres de la commune de Saint-Germain-en-Laye peut sembler trivial, mais en creusant un peu, on comprend qu'il s'agit de créer des outils pour répondre à des questions liées aux risques climatiques. Les arbres ne sont pas seulement des éléments décoratifs en ville ; ce sont des acteurs clés dans la lutte contre le réchauffement climatique, l'atténuation des pics thermiques en milieu urbain, et l'amélioration de la qualité de vie des habitants. Par ailleurs, ces bénéfices dépendent largement de la diversité, de la santé et de la répartition des arbres au sein de l'écosystème urbain. En établissant **une relation**

binaire basée sur des critères comme **la hauteur et le diamètre** des arbres, ainsi que sur les notions de **proximité et éloignement**, on peut mieux comprendre leurs interactions écologiques et leur résilience face aux événements climatiques extrêmes. Par exemple, en identifiant des **clusters** d'arbres homogènes ou vulnérables, les autorités locales peuvent anticiper des stratégies de gestion plus adaptées : renforcer les arbres exposés aux vents forts ou encore diversifier les plantations pour limiter les risques de pertes massives. Notre projet concerne donc l'analyse de 709 arbres situés dans la commune de Saint-Germain-en-Laye.

2.2.1 Pourquoi créer une relation binaire : $Q = I_Q + P_Q$?

Créer la relation binaire Q nous permet de structurer et d'organiser les données des arbres en fonction de leurs caractéristiques mesurables (hauteur et diamètre) un peu comme si on faisait du **"clustering"**. La relation Q est définie comme la somme de deux sous-relations :

— l_Q : indique si deux arbres sont **proches** en termes de hauteur et diamètre.
— p_Q : indique si deux arbres sont **éloignés** en termes de hauteur et diamètre.

La relation binaire divise l'ensemble des arbres en **groupes** facilement interprétables (proches ou éloignés).En reliant les arbres selon leurs proximités ou différences, on peut obtenir une représentation graphique de leur interaction, utile pour détecter des clusters.

Formellement, une relation binaire Q sur un ensemble des arbres $A = \{a_1, a_2, \ldots, a_{709}\}$ où chaque arbre a_i est caractérisé par deux variables : $h(a_i)$ (la hauteur de l'arbre i), $d(a_i)$ (le diamètre de l'arbre i) est définie comme un sous-ensemble de $A \times A$, tel que $Q \subseteq A \times A$. Dans ce projet, chaque élément de A correspond à un arbre de la base de données, et la relation Q est composée de deux sous-relations :

$$Q = I_Q + P_Q$$

2.2.2 Définition de la notion de proximité

La notion de proximité est essentielle pour caractériser les relations entre les arbres. Elle nous permet de quantifier à quel point deux arbres partagent des caractéristiques similaires en termes de **hauteur et diamètre**. Pour ce faire, nous allons utiliser la définition de proximité suggérée par le professeur Marc-Arthur Diaye.

Posons : $h(a_i) = h(i)$, $h(a_j) = h(j)$ et $d(a_i) = d(i)$, $d(a_j) = d(j)$

1. La **relation de proximité** I_Q se résume comme suit :

$$\text{Arbre } i \, I_Q \, \text{Arbre } j \iff h(j) - s_1 \leq h(i) \leq h(j) + s_1 \quad \text{et} \quad d(j) - s_2 \leq d(i) \leq d(j) + s_2$$

2. La **relation de distance** P_Q se résume comme suit :

$$\text{Arbre } i \, P_Q \, \text{Arbre } j \iff h(i) > h(j) + s_1 \quad \text{et} \quad d(i) > d(j) + s_2$$

Ici, s_1 et s_2 sont des seuils que nous définirons ultérieurement.

2.2.3 Analyse Exploratoire des Données : sgl-arbres-urbains-wgs84

Avant de passer à tout autre chose, nous allons d'abord préparer les données afin de pouvoir les utiliser. Nous allons d'abord essayer de voir si nos variables **hauteur** et **diamètre** contiennent des valeurs manquantes. Si oui, alors on les remplacera par des moyennes.

```
> any(is.na(hauteur))
[1] FALSE
> any(is.na(diametre))
[1] FALSE
```

Nous pouvons constater que nos variables hauteur et diamètre n'ont aucune valeur manquante.

Passons donc à l'analyse descriptive de nos 2 variables

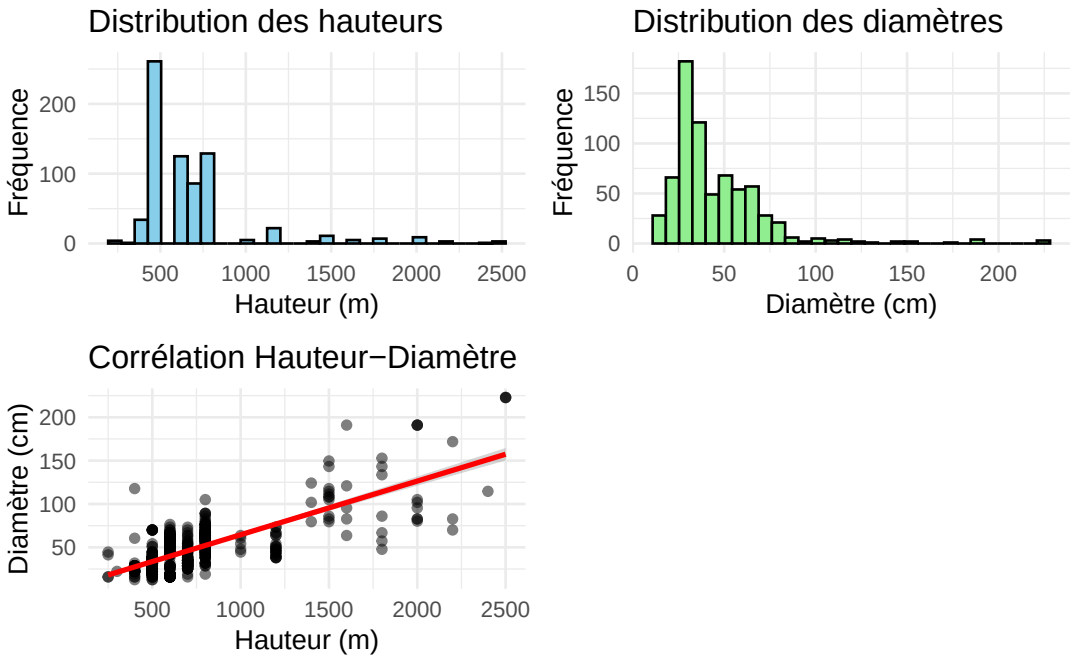


FIGURE 11 – Analyse des hauteurs et diamètres des arbres

Nous observons que la distribution des hauteurs montre une concentration majoritaire entre 500-1000 m, avec un pic autour de 500 m. La répartition des diamètres présente une distribution asymétrique positive, avec la plupart des arbres ayant un diamètre entre 20-80 cm, typique d'une forêt naturelle. Le graphique de corrélation hauteur-diamètre révèle une relation positive linéaire (R^2 d'environ 0.6), ce qui confirme que les arbres plus hauts ont généralement des diamètres plus importants, bien qu'avec une dispersion notable autour de la tendance.

La moyenne de la hauteur des arbres est de 692.5 m avec un écart-type de 333.02 m, ce qui indique une grande variabilité absolue dans les tailles. En comparaison, le diamètre a une moyenne de 45.52 cm et un écart-type de 27.14 cm, ce qui reflète une dispersion relative plus marquée par rapport à sa moyenne. Cela montre que la hauteur varie sur une échelle beaucoup plus large que le diamètre, mais le diamètre présente une variabilité proportionnelle plus importante.

TABLE 8 – Résumé statistique des variables numériques

Statistiques	Hauteur (m)	Diamètre (cm)
Minimum	250.0	12.73
1er Quartile	500.0	28.65
Médiane	600.0	38.20
Moyenne	692.5	45.52
3e Quartile	800.0	57.30
Maximum	2500.0	222.82
Écart-type	333.0233	27.14145

2.2.4 Définition des Seuils : s_1 et s_2

De façon générale, le seuil ici n'est qu'une limite ou une **"marge"** que nous allons fixer pour décider si deux arbres sont similaires (**proches**) ou différents (**éloignés** en fonction de leur hauteur et de leur diamètre.

Dans une relation de proximité :

Le s_1 (seuil de la hauteur) est une marge qu'on ajoute ou qu'on retire par exemple à la hauteur d'un arbre (i) pour voir si un autre arbre (j) a une hauteur proche de lui, en se basant sur la relation de proximité (I_Q). Par exemple, si un arbre mesure 10 mètres et que le seuil est 2, on considère qu'un autre arbre avec une hauteur entre 8 et 12 mètres est "proche" en termes de hauteur. C'est pareil pour s_2 (le diamètre). Si le seuil est 5 cm et un arbre a un diamètre de 30 cm, on dira qu'un autre arbre avec un diamètre entre 25 et 35 cm est "proche" en termes de diamètre. Et si, **Les deux conditions sont vérifiées conjointement on pourra ainsi conclure que oui : L'arbre (i) est à proximité de l'arbre (j)**

Dans une relation de distance ou d'éloignement :

Le s_1 est un seuil minimal de différence de hauteur nécessaire pour que l'arbre i soit considéré comme éloigné de l'arbre j, selon la relation P_Q. Par exemple, si un arbre mesure 10 mètres et que seuil est 2, il sera considéré comme éloigné d'un autre arbre seulement si sa hauteur est strictement supérieure à 12 mètres. De même pour s_2 (diamètre), si un arbre a un diamètre de 30 cm et que le seuil est 5 cm, il sera éloigné d'un autre arbre seulement si son diamètre est strictement supérieur à 35 cm. Et si, **les deux conditions d'éloignement sont vérifiées simultanément, on pourra alors conclure que oui : L'arbre i est distant ou éloigné de l'arbre j.**

2.2.5 Comment déterminer les Seuils (s_1 et s_2) ?

Pour déterminer les seuils de s_1 et de s_2), dans notre étude des relations de proximité entre les arbres de Saint-Germain-en-Laye, nous avons choisi une méthode basée sur les quantiles. Pourquoi ce choix ? Parce qu'il s'avère plus adapté à notre objectif : comprendre les relations entre les arbres en se basant sur leurs différences physiques. Les quantiles, et plus précisément les quartiles (25%, 50%, 75%), permettent de segmenter et d'organiser les différences de manière plus intuitive. Par exemple le premier quartile (25%) comme référence, pourra permettre d'identifier de manière naturelle les paires d'arbres qui présentent les différences les plus faibles. Quoiqu'il soit possible de passer par les écart-types des variables aussi.

L'output des seuils de proximité et d'éloignement des arbres : quantiles des différences"

```
> resultats <- analyser_differences(arbres)
Analyse pour la relation de proximite I :
Quartiles des differences absolues de hauteur (m):
  25%   50%   75%
  100   200   300

Quartiles des differences absolues de diametre (cm):
      25%       50%       75%
  6.366198 19.098593 35.014087

Analyse pour la relation deloignement P :
Quartiles des differences positives de hauteur (m):
  25%   50%   75%
  100   200   400

Quartiles des differences positives de diametre (cm):
      25%       50%       75%
  9.549297 19.098593 35.014087
```

Détermination des seuils pour les relations de proximité (I) et d'éloignement (P)

1. Seuils pour la relation de proximité (I)

Pour caractériser la proximité entre deux arbres, nous choisissons le premier quartile (25%) des différences absolues. Avec Q_1, nous définissons comme "proches" les arbres dont les différences de caractéristiques se situent parmi les 25% les plus faibles observées et nous assurons une définition stricte de la proximité. Ce qui nous garantit que seules les paires d'arbres présentant des différences vraiment minimales seront considérées comme proches. On a donc :

- $s_{1\_I} = 100$ **mètres pour la hauteur**
- $s_{2\_I} = 6.37$ **cm pour le diamètre**

2. Seuils pour la relation d'éloignement (P)

Pour caractériser l'éloignement, nous optons pour le troisième quartile (75%) des différences positives. Avec Q_3, nous considérons comme "éloignés" les arbres dont les différences se situent parmi les 25% les plus élevées des différences positives. Ce qui nous assure que la relation P caractérise des différences significatives et non marginale. On a donc :

- $s_{1\_P} = 400$ **mètres pour la hauteur**
- $s_{2\_P} = 35.01$ **cm pour le diamètre**

2.2.6 Validation de nos Seuils définis

```
> resultats_validation <- valider_relations_modifiee(arbres, seuils_proximite,
    seuils_eloignement)
Validation des seuils choisis :

Seuils de proximite (I) :
s_1_I (hauteur) = 100 metres
s_2_I (diametre) = 6.37 cm

Seuils deloignement (P) :
```

```
s_1_P (hauteur) = 400 metres
s_2_P (diametre) = 35.01 cm

Statistiques globales :
Nombre total de paires darbres : 250986

Relations de proximite (I) :
Nombre de paires proches : 39971.5
Pourcentage : 15.93 %

Relations deloignement (P) :
Nombre de paires eloignees : 14010.5
Pourcentage : 5.58 %
```

Pour un total de n = 709 arbres , nous avons :

$$\text{Nombre de paires} = \binom{n}{2} = \frac{n \times (n-1)}{2}$$

$$\text{Nombre de paires} = \frac{709 \times (709-1)}{2}$$

$$\text{Nombre de paires} = \frac{709 \times 708}{2}$$

$$\text{Nombre de paires} = \frac{502572}{2}$$

$$\text{Nombre de paires} = 250986$$

Ainsi, Sur un total de **250986 paires d'arbres possibles**, nos seuils ont permis de caractériser deux types de relations distinctes, et de laisser une zone intermédiaire substantielle qui n'appartient à aucune des deux catégories.

Pour la relation de proximité (I)

Avec nos seuils de $s_1\_I = 100$ mètres pour la hauteur et $s_2\_I = 6.37$ cm pour le diamètre, environ 39971 paires d'arbres (15.93% du total) sont considérées comme proches. Ce qui est tout à fait correct puisqu'il indique que nous avons identifié une proportion significative mais sélective de paires d'arbres présentant des caractéristiques vraiment similaires.

Pour la relation d'éloignement (P)

Les seuils d'éloignement ($s_1\_P = 400$ mètres et $s_2\_P = 35.01$ cm) ont identifié 14,010 paires d'arbres (5.58% du total) comme étant éloignées. Ce pourcentage plus faible est cohérent avec notre approche plus restrictive pour la relation d'éloignement, qui nécessite des différences importantes dans les deux dimensions simultanément.

Zone intermédiaire

Par déduction, environ 78.49% des paires d'arbres se trouvent dans **une zone intermédiaire**,car ni particulièrement proches ni particulièrement éloignées. Ce qui est tout à fait naturel pour une classification des relations entre arbres urbains. Par conséquent, nos seuils sont validés. Nous pouvons donc passer à la création de la relation binaire.

2.2.7 Construction de la relation binaire : $Q = I_Q + P_Q$

```
> relation_Q <- construire_relation_Q(arbres, seuils_proximite, seuils_
    eloignement)
Proprietes de la relation binaire Q :
Dimension de la matrice : 709 x 709
Nombre total de relations (I(Q) + P(Q)) : 107964
Nombre de relations de proximite (I(Q)) : 79943
Nombre de relations deloignement (P(Q)) : 28021
```

Analyse

La matrice Q obtenue est de dimension 709 x 709, ce qui correspond au nombre total d'arbres dans notre jeu de données. Cette matrice carrée permet de représenter toutes les relations possibles entre chaque paire d'arbres. Nous observons aussi que le nombre total de relations ($I_Q + P_Q$) s'élève à 107964, et se décompose en 79943 de relations de proximité (I_Q) et 28021 de relations d'éloignement (P_Q). Cette distribution nous montre que **les relations de proximité sont environ trois fois plus nombreuses que les relations d'éloignement**, ce qui peut s'expliquer par deux facteurs :

— Premièrement, la relation de proximité (I_Q) est symétrique : si l'arbre i est proche de l'arbre j, alors j est nécessairement proche de i. Cela double naturellement le nombre de relations de proximité.
— Deuxièmement, la relation d'éloignement (P_Q) est plus restrictive dans sa définition, ce qui nécessite le fait qu'un arbre soit strictement plus grand et plus large qu'un autre d'un certain seuil minimal prédéfini. Cette condition plus stricte explique le nombre plus faible de relations d'éloignement.

2.2.8 Visualisation de la Relation binaire ; proximité et éloignement

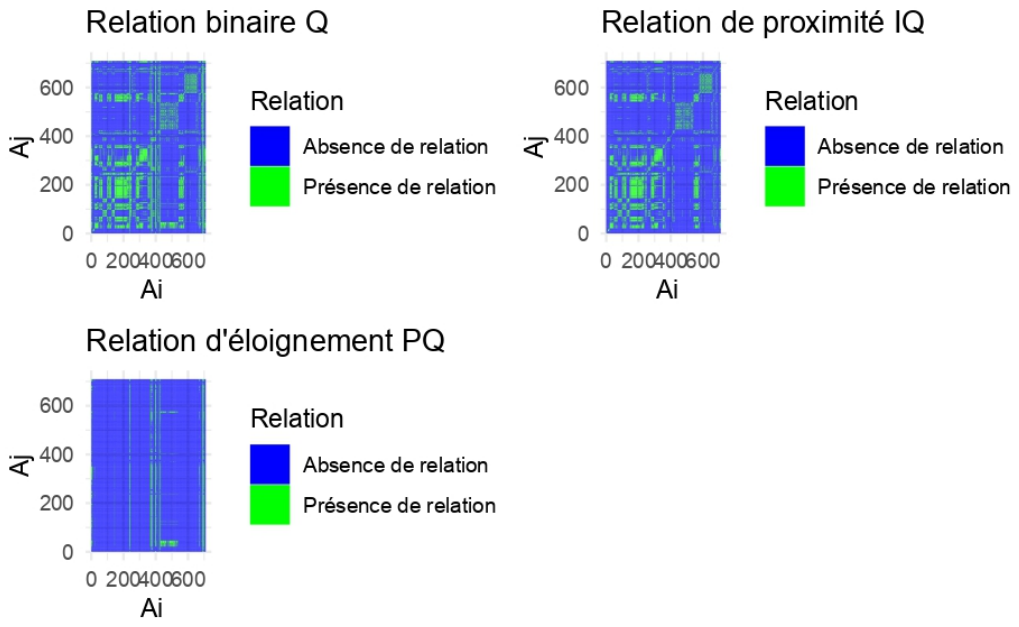


FIGURE 12 – Représentation des relations

2.2.9 Fermeture Transitive de Q

La **fermeture transitive** de Q, notée Q^+, est la plus petite relation transitive contenant Q. En termes de graphes, cela revient à considérer les chemins reliant deux nœuds ou arbres dans un graphe orienté où les arcs sont définis par Q. Pour une relation Q sur un ensemble A, Q^+ est définie comme :

$$Q^+ = Q + Q^2 + Q^3 + \cdots$$

où Q est la matrice d'adjacence associée à la relation Q, et Q^k représente les chemins de longueur k.

Ici, la fermeture transitive permet d'établir des chaînes de dominance à travers la relation de distance P_Q, : si l'arbre i domine l'arbre j (iP_Qj) et j domine k (jP_Qk), alors i domine indirectement k ($iP_{Q^+}k$).

```
Application a notre relation Q
Q_star <- calculer_fermeture_transitive(relation_Q$Q)

Proprietes de la fermeture transitive :
Nombre de relations dans Q initiale : 107964
Nombre de relations dans la fermeture transitive : 452035
Nombre de nouvelles relations ajoutees : 344071
```

Par ailleurs, une fermeture transitive doit posséder trois propriétés clés : la réflexivité, la transitivité et être une extension de la relation initiale. Ainsi, nous devons donc démontrer cela pour confirmer notre fermeture transitive.

2.2.10 Analyse de la Fermeture Transitive

Analysons les résultats de la fermeture transitive de notre relation binaire Q :

État Initial

La relation binaire Q comportait initialement 107964 relations, ce qui représentait la somme de nos relations de proximité (I_Q) et d'éloignement (P_Q).

Fermeture Transitive

Après l'application de l'algorithme (Warshall), la fermeture transitive Q^+ contient 452035 relations. Cette augmentation significative s'explique par l'ajout de 344071 nouvelles relations transitives.

Interprétation

Cette augmentation importante (plus que quadruple) du nombre de relations indique que de nombreuses connexions indirectes existaient entre les arbres. Par exemple, si l'arbre i était en relation avec j, et j avec k, la fermeture transitive a ajouté une relation entre i et k. Le fait que le nombre de relations ait augmenté de manière aussi substantielle suggère que notre relation initiale Q contenait de nombreuses chaînes de relations qui ont été complétées par la fermeture transitive.

2.2.11 Vérification de la Fermeture

Nous pouvons vérifier les propriétés mathématiques de cette fermeture transitive (réflexivité, transitivité, extension) pour confirmer que notre résultat satisfait bien toutes les conditions requises.

```
> verifier_proprietes_fermeture(relation_Q$Q, Q_star)

Verification des proprietes de la fermeture transitive
Reflexivite : TRUE
Transitivite : TRUE
Extension de la relation initiale : TRUE
```

ces résultats confirment donc la validation mathématique de notre fermeture transitive. Ainsi, la réflexivité confirme par exemple que chaque arbre est en relation avec lui-même dans la fermeture transitive. Dans le contexte de notre étude, cela signifie que pour tout arbre i , i est en relation avec lui-même dans Q^+. la transitivité elle-même nous indique que pour tous les arbres i, j et k, si i est en relation avec j et j est en relation avec k, alors i est en relation avec k dans Q^+. La propriété de l'extension nous confirme que la fermeture transitive Q^+ conserve toutes les relations présentes dans la relation initiale Q, tout en y ajoutant les nouvelles relations transitives. Autrement dit, nous n'avons perdu aucune des relations originales lors de la construction de la fermeture transitive.

Notre implémentation a donc réussi à créer une structure relationnelle complète et cohérente entre les arbres de Saint-Germain-en-Laye.

2.2.12 Visualisation de la Fermeture Transitive

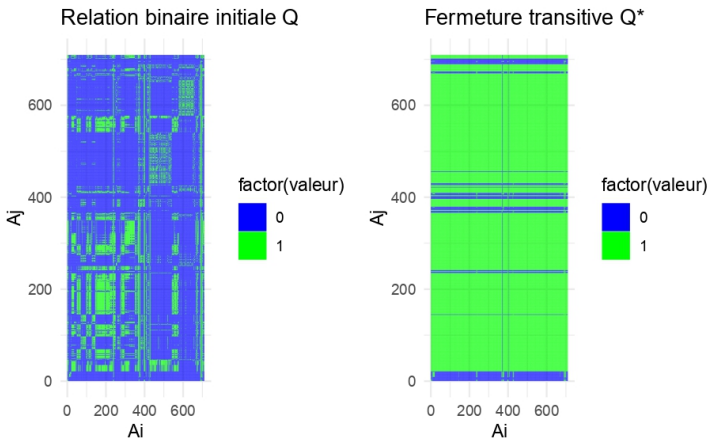


FIGURE 13 – Comparaison entre la relation binaire initiale Q et sa fermeture transitive Q^+

Ce graphique met en évidence l'extension des relations dans la fermeture transitive Q^+. Dans la matrice initiale Q, les relations directes sont localisées (zones vertes), tandis que la fermeture transitive Q^+ ajoute de nombreuses relations supplémentaires (augmentation significative des zones vertes). Cela illustre clairement l'effet de la transitivité, où chaque chaîne de relations indirectes est complétée pour assurer une connectivité globale.

2.3 Partie 3 : OPTIMISATION SOUS R

L'objectif de cette partie de notre projet, est de résoudre un problème d'optimisation mathématique défini par une fonction objectif $f(x, y, z)$ et un ensemble de contraintes. Imaginons que nos variables x, y, et z représentent des investissements dans trois projets différents, avec un budget total limité. La fonction $f(x, y, z)$ pourrait représenter un coût total que nous cherchons à minimiser tout en respectant les contraintes budgétaires et les niveaux minimums d'investissement nécessaires pour que chaque projet soit viable. Cette partie est subdivisée en 2 sous-parties. Dans la première, nous irons à la résolution du problème par **la méthode du Lagrangien** étudiée en cours, avec des explications à l'appui. Dans la seconde, nous résoudrons ce problème d'optimisation sous R tout en prenant le soin d'expliquer les codes.

2.3.1 Résolution à la Main

Notre résolution porte sur l'optimisation d'une fonction objectif **quadratique** $f(x, y, z) = x^2 + y^2 + z^2 - 2x - 3y + z$ soumise à un ensemble de contraintes linéaires définissant un domaine D dans $\mathbb{R}^3_+$. Pour ce faire, nous procéderons à la résolution du problème en 5 étapes. La première étape consistera à poser le Lagrangien du problème. Dans la deuxième,nous déterminerons l'ensemble sur lequel le problème est défini et ainsi vérifier la condition de qualification. Puis ensuite, nous déterminerons les points critiques dans l'étape 3 et voir à quoi ressemblent les minima dans l'étape 4. La dernière partie consistera à vérifier la globalité des extrema avec la matrice hessienne bordée.

2.3.2 Optimisation : Maximisation ou Minimisation ?

Soit la fonction $f(x, y, z) = x^2 + y^2 + z^2 - 2x - 3y + z$. Les termes quadratiques $x^2, y^2, z^2 \geq 0$ dominent la fonction pour $x, y, z \to +\infty$, entraînant :

$$\lim_{\|(x,y,z)\| \to +\infty} f(x, y, z) = +\infty.$$

Les termes linéaires $-2x, -3y, +z$ augmentent ou diminuent proportionnellement à x, y, z, mais leur croissance est beaucoup plus lente comparée à celle des termes quadratiques x^2, y^2, z^2. Par conséquent, ces termes linéaires ne peuvent compenser la croissance rapide des termes quadratiques.

Dans ce problème, l'ensemble

$$D = \{(x, y, z) \in \mathbb{R}^3 : x + 2y + z \leq 4,\ x + y \geq 1,\ x, y, z \geq 0\}$$

est **borné et fermé** parce que les contraintes imposent des limites supérieures sur x, y, z et sont définies par des inégalités $(\leq, \geq)$ continues. Par conséquent,l'ensemble $D = \{(x, y, z) \in \mathbb{R}^3 : x + 2y + z \leq 4, x + y \geq 1, x, y, z \geq 0\}$ est **compact**. La compacité garantit l'existence d'un minimum pour une fonction continue selon les propriétés asymptotiques d'une fonction.

Ainsi, $f(x, y, z)$ est uniquement optimisable par *minimisation*.

Étape 1 : Le Lagrangien

Soit la fonction suivante : $f(x, y, z) = x^2 + y^2 + z^2 - 2x - 3y + z$ sur $D = \{(x, y, z) \in \mathbb{R}^3 : x + 2y + z \leq 4, x + y \geq 1, x, y, z \geq 0\}$.

Pour simplifier l'analyse, nous allons nous restreindre à $\mathbb{R}^3_+$, et donc négliger les contraintes de positivité puisque nos variables doivent être positives. Nous nous concentrerons donc sur les deux contraintes principales dans le domaine $D = \{(x, y, z) \in \mathbb{R}^3_+\}$.

Le Lagrangien de la fonction $f(x, y, z) = x^2 + y^2 + z^2 - 2x - 3y + z$, soumis aux contraintes

$$g_1(x, y, z) = x + 2y + z - 4 \leq 0, \quad g_2(x, y, z) = -x - y + 1 \leq 0,$$

et avec les multiplicateurs de Lagrange $\lambda_1 \geq 0$ et $\lambda_2 \geq 0$, s'écrit :

$$\mathcal{L}(x, y, z, \lambda_1, \lambda_2) = f(x, y, z) + \lambda_1 g_1(x, y, z) + \lambda_2 g_2(x, y, z).$$

En remplaçant les expressions de notre fonction f et des contraintes g_1, et g_2, on a :

$$\mathcal{L}(x, y, z, \lambda_1, \lambda_2) = x^2 + y^2 + z^2 - 2x - 3y + z + \lambda_1(x + 2y + z - 4) + \lambda_2(-x - y + 1).$$

Étape 2 : Vérification des conditions de qualification

Vérifier la condition de qualification, consisterait à s'assurer que les contraintes définissent un ensemble D régulier et que les gradients des fonctions contraintes ne sont pas colinéaires au point candidat où les contraintes sont actives.

Nous avons : $g(x, y, z) = (x + 2y + z - 4, -x - y + 1)$

Nous pouvons reformuler les contraintes comme suit :

$$\begin{cases} g_1(x, y, z) = x + 2y + z - 4 \leq 0, \\ g_2(x, y, z) = -x - y + 1 \leq 0. \end{cases}$$

Déterminons : les gradients des contraintes ∇g_2 et ∇g_1 afin de créer la matrice jacobienne de g

Gradient de $g_1(x, y, z)$:

$$\nabla g_1(x, y, z) = \left(\frac{\partial g_1}{\partial x}, \frac{\partial g_1}{\partial y}, \frac{\partial g_1}{\partial z} \right)$$
$$\nabla g_1(x, y, z) = (1, 2, 1)$$

Gradient de $g_2(x, y, z)$:

$$\nabla g_2(x, y, z) = \left(\frac{\partial g_2}{\partial x}, \frac{\partial g_2}{\partial y}, \frac{\partial g_2}{\partial z} \right)$$
$$\nabla g_2(x, y, z) = (-1, -1, 0)$$

Matrice de la Jacobienne de g :

$$J_g = \begin{pmatrix} \frac{\partial g_1}{\partial x} & \frac{\partial g_1}{\partial y} & \frac{\partial g_1}{\partial z} \\ \frac{\partial g_2}{\partial x} & \frac{\partial g_2}{\partial y} & \frac{\partial g_2}{\partial z} \end{pmatrix}$$

On a donc :

$$J_g = \begin{pmatrix} 1 & 2 & 1 \\ -1 & -1 & 0 \end{pmatrix} \implies \textbf{Rang}(J_g) = 2$$

Le rang de la matrice jacobienne des contraintes est égal au nombre de contraintes indépendantes, soit **2** car les lignes sont linéairement indépendantes, donc la condition de qualification de rang constant est satisfaite. **Le problème est bien défini sur $\mathcal{D} \subset \mathbb{R}^3_+$.**

Étape 3 : Détermination des points critiques

Le Lagrangien est donné par :

$$\mathcal{L}(x, y, z, \lambda_1, \lambda_2) = x^2 + y^2 + z^2 - 2x - 3y + z + \lambda_1(x + 2y + z - 4) + \lambda_2(-x - y + 1),$$

où $\lambda_1 \geq 0$ et $\lambda_2 \geq 0$.

Le système des **conditions nécessaires du premier ordre** se réécrit comme suit :

$$\begin{cases} \frac{\partial L}{\partial x} = 0 \\ \frac{\partial L}{\partial y} = 0 \\ \frac{\partial L}{\partial z} = 0 \\ \lambda_1(x + 2y + z - 4) = 0 \\ \lambda_2(-x - y + 1) = 0 \end{cases} \Rightarrow \begin{cases} 2x - 2 + \lambda_1 - \lambda_2 = 0 & (1) \\ 2y - 3 + 2\lambda_1 - \lambda_2 = 0 & (2) \\ 2z + 1 + \lambda_1 = 0 & (3) \\ \lambda_1(x + 2y + z - 4) = 0 & (4) \\ \lambda_2(-x - y + 1) = 0 & (5) \end{cases}$$

Selon les valeurs des multiplicateurs de Lagrange λ_1 et λ_2, nous pouvons faire l'analyse selon quatre cas possibles : $\lambda_1 \neq 0$ et $\lambda_2 = 0$, $\lambda_1 = 0$ et $\lambda_2 \neq 0$, $\lambda_1 = 0$ et $\lambda_2 = 0$, $\lambda_1 \neq 0$ et $\lambda_2 \neq 0$

Cas 1 : $\lambda_1 \neq 0$ et $\lambda_2 = 0$

Dans ce cas, d'après (4), nous avons $x + 2y + z - 4 = 0$. Le système à résoudre devient :

$$2x - 2 + \lambda_1 = 0 \ (1'), \quad 2y - 3 + 2\lambda_1 = 0 \ (2'), \quad 2z + 1 + \lambda_1 = 0 \ (3), \quad \text{et} \quad x + 2y + z - 4 = 0 \ (4').$$

De (3), on tire $z = -(\lambda_1 + 1)/2$. De (1), on tire $x = (-\lambda_1 + 2)/2$. De (2), on tire $y = (-2\lambda_1 + 3)/2$.

En substituant ces expressions dans (4), on obtient :

$$(-\lambda_1 + 2)/2 + 2 \cdot (-2\lambda_1 + 3)/2 - (\lambda_1 + 1)/2 = 4.$$

Après simplification, cela donne $-\lambda_1 - 2\lambda_1 - 1/2 = 0$, soit $-3\lambda_1 = 1/2$. Par conséquent, $\lambda_1 = -1/6$.

En remplaçant $\lambda_1 = -1/6$ dans les expressions de x, y, et z, on obtient :

$$x = 11/12, \quad y = 5/3, \quad z = -5/12.$$

Nous n'avons pas besoin de démontrer si les contraintes principales sont satisfaites. Pourquoi ? Parce que la condition $z \geq 0$ et la condition $\lambda_1 \geq 0$ ne sont pas satisfaites car $z = -5/12 < 0$ et $\lambda_1 = -1/6 < 0$. Par conséquent, le point $\left(\frac{11}{12}, \frac{5}{3}, -\frac{5}{12}, -\frac{1}{6}, 0\right)_{(x,y,z,\lambda_1,\lambda_2)}$ n'est pas admissible comme point critique car nous travaillons sur $\mathbb{R}^3_+$.

Cas 2 : $\lambda_1 = 0$ et $\lambda_2 \neq 0$,

Dans ce cas, d'après (5), nous avons $-x - y + 1 = 0$. Le système à résoudre devient :

$$2x - 2 - \lambda_2 = 0 \ (1"), \quad 2y - 3 - \lambda_2 = 0 \ (2"), \quad 2z + 1 = 0 \ (3'), \quad -x - y + 1 = 0 \ (5").$$

De (3'), on tire $z = -1/2$. De (1"), on tire $x = 1 + \lambda_2/2$. De (2"), on tire $y = (3 + \lambda_2)/2$.

En substituant ces expressions dans (5"), on obtient :

$$(-1 - \lambda_2/2) - (3/2 + \lambda_2/2) - 1 = 0.$$

Après simplification :

$$-3/2 - \lambda_2 = 0 \implies \lambda_2 = -3/2$$

En remplaçant $\lambda_2 = 3/2$ dans les expressions de x, y, et z, on obtient :

$$x = 1 - 3/4 = 1/4, \quad y = 3/2 - 3/4 = 3/4, \quad z = -1/2.$$

De même que le cas 1, ce point n'est pas un point critique car $z < 0$, ce qui viole la contrainte $z \geq 0$.

Cas 3 : $\lambda_1 = 0$ et $\lambda_2 = 0$

Le système devient donc :

$$2x - 2 = 0, \quad 2y - 3 = 0, \quad 2z + 1 = 0.$$

Ce qui donne directement :

$$x = 1, \quad y = 3/2, \quad z = -1/2.$$

Ce point n'est donc pas un point critique car $z < 0$, ce qui viole la contrainte $z \geq 0$.

Cas 4 : $\lambda_1 \neq 0$ et $\lambda_2 \neq 0$

Dans ce cas, l'équation (4) et (5) devient :

$$\begin{cases} x + 2y + z - 4 = 0 \quad (\text{puisque } \lambda_1 \neq 0), \\ -x - y + 1 = 0 \quad (\text{puisque } \lambda_2 \neq 0). \end{cases}$$

De $(-x - y + 1 = 0)$, on obtient : $x + y = 1 \implies y = 1 - x$.

De $x + 2y + z = 4$, en remplaçant y par $1 - x$, on trouve :

$$x + 2(1 - x) + z = 4 \implies x + 2 - 2x + z = 4 \implies z = 4 - (2 - x) = x + 2.$$

L'équation (3) donne :

$$2z + 1 + \lambda_1 = 0 \implies 2(x + 2) + 1 + \lambda_1 = 0 \implies 2x + 4 + 1 + \lambda_1 = 0 \implies \lambda_1 = -2x - 5.$$

Substituons $\lambda_1 = -2x - 5$ dans (1) et (2) :

Pour (1) :

$$2x - 2 + \lambda_1 - \lambda_2 = 0 \implies -2 + 2x + (-2x - 5) - \lambda_2 = 0 \implies -7 - \lambda_2 = 0 \implies \lambda_2 = -7.$$

Pour (2) :

$$2y - 3 + 2\lambda_1 - \lambda_2 = 0.$$

Or $y = 1 - x$, donc $2y - 3 = 2(1 - x) - 3 = 2 - 2x - 3 = -1 - 2x$. Et $2\lambda_1 = 2(-2x - 5) = -4x - 10$.

On obtient alors :

$$(-1 - 2x) + (-4x - 10) - \lambda_2 = 0 \implies -11 - 6x - \lambda_2 = 0.$$

Comme on a déjà trouvé $\lambda_2 = -7$, on pose :

$$-11 - 6x + 7 = 0 \implies -4 - 6x = 0 \implies x = -2/3.$$

On remonte alors à y et z :

$$y = 1 - x = 1 - \left(-\frac{2}{3}\right) = 1 + \frac{2}{3} = \frac{5}{3}, \quad z = x + 2 = -\frac{2}{3} + 2 = \frac{4}{3}.$$

Enfin,

$$\lambda_1 = -2x - 5 = -2\left(-\frac{2}{3}\right) - 5 = \frac{4}{3} - 5 = -\frac{11}{3}, \quad \lambda_2 = -7.$$

On vérifie que ces valeurs satisfont bien toutes les équations et que $\lambda_1 \neq 0$, $\lambda_2 \neq 0$. La solution (unique) dans ce cas est donc :

$$x = -\frac{2}{3}, \quad y = \frac{5}{3}, \quad z = \frac{4}{3}, \quad \lambda_1 = -\frac{11}{3}, \quad \lambda_2 = -7$$

. Ce point n'est certainement pas un point critique car ne respecte pas la contrainte de non-négativité, $x < 0$.

La conjecture :

Nous avons examiné les différents « cas » (par exemple $\lambda_1 = 0$ ou $\lambda_1 \neq 0$, etc.), ce qui revient à tester si la contrainte est strictement satisfaisante ($<$) ou bien saturée ($=$). **Pourquoi cela ne nous a-t-il pas suffit à déterminer ce fameux point optimal ?**

Dans notre problème d'optimisation, nous avons eu à négliger dans le traitement du KKT les multiplicateurs pour $x \geq 0, y \geq 0, z \geq 0$ pour simplifier la résolution. Puisque affecter des multiplicateurs du Lagrangien à ces derniers fait beaucoup de cas à vérifier. Or, ce que nous constatons, est que l'optimum s'il ne peut être dans l'intérieur strict **(car le point qui annule le gradient est hors du domaine)**, donc il **«doit»** se trouver quelque part sur la frontière (intersection des contraintes). Parce que les **bornes** $x = 0, y = 0, z = 0$ elles-mêmes sont autant de « contraintes » susceptibles d'être actives ou non.

Alors ce que nous faisons, au lieu de traiter exhaustivement tous les multiplicateurs, on peut directement examiner les frontières.

Examen 1 : $x = 0$

Contraintes restantes : $2y + z \leq 4,\ y \geq 1,\ y, z \geq 0.$

Fonction à minimiser devient : $f(0, y, z) = y^2 + z^2 - 3y + z.$

Recherche de point stationnaire ($y > 0, z > 0$) : $\frac{\partial f}{\partial y} = 2y - 3 = 0 \implies y = 1.5,\ \frac{\partial f}{\partial z} = 2z + 1 = 0 \implies z = -0.5$ (impossible car $z \geq 0$). Donc, pas de point stationnaire intérieur.

Frontière $z = 0$: On minimise $f(0, y, 0) = y^2 - 3y$ avec $1 \leq y \leq 2$. Minimum à $y = 1.5$: $f(0, 1.5, 0) = -2.25$.

Frontière $z = 4 - 2y$: En pratique, f donne des valeurs supérieures à -2.25.

Le point optimal lorsque $x = 0$ est $(0, 1.5, 0)$ avec $f = -2.25$.

Examen 2 : $y = 0$

Contraintes restantes : $x + z \leq 4,\ x \geq 1,\ x, z \geq 0.$

Fonction à minimiser devient : $f(x, 0, z) = x^2 + z^2 - 2x + z.$

Recherche de point stationnaire ($x > 0, z > 0$) :

$$\frac{\partial f}{\partial x} = 2x - 2 = 0 \implies x = 1, \quad \frac{\partial f}{\partial z} = 2z + 1 = 0 \implies z = -0.5 \text{ (impossible car } z \geq 0).$$

Donc, pas de point stationnaire intérieur admissible.
Frontière $z = 0$: $f(x, 0, 0) = x^2 - 2x \quad$ avec $\quad 1 \leq x \leq 4$.

Minimum sur $[1, 4]$ atteint en $x = 1$: $f(1, 0, 0) = 1 - 2 = -1$.

Le point optimal lorsque $y = 0$ est $(1, 0, 0)$ avec $f = -1$.

Examen 3 : $z = 0$

Contraintes restantes : $x + 2y \leq 4,\ x + y \geq 1,\ x, y \geq 0$.

Fonction à minimiser devient : $f(x, y, 0) = x^2 + y^2 - 2x - 3y$.

Point stationnaire intérieur $(x > 0, y > 0)$:

$$\frac{\partial f}{\partial x} = 2x - 2 = 0 \implies x = 1, \quad \frac{\partial f}{\partial y} = 2y - 3 = 0 \implies y = 1.5.$$

Vérifions si $(1, 1.5)$ satisfait les contraintes :

$$x + 2y = 1 + 2 \cdot 1.5 = 4 \text{ (ok, contrainte saturée)}, \quad x + y = 1 + 1.5 = 2.5 \geq 1, \quad x, y \geq 0.$$

Et oui : $(1, 1.5)$ est bien admissible. On obtient alors :

$$f(1, 1.5, 0) = 1^2 + (1.5)^2 - 2 \cdot 1 - 3 \cdot 1.5 = 1 + 2.25 - 2 - 4.5 = -3.25.$$

C'est un candidat très intéressant.

Comparaison avec les autres cas : Pour $x = 0$, on avait trouvé un minimum $(0, 1.5, 0)$ avec $f = -2.25$.
Pour $y = 0$, on avait $(1, 0, 0)$ avec $f = -1$.

Conclusion pour $z = 0$: Le point $(x, y, z) = (1, 1.5, 0)$ fournit $f = -3.25$, valeur plus petite (donc meilleure) que les minima trouvés dans les cas $x = 0$ ou $y = 0$.

Notre point optimal est donc :

$$\boxed{(x, y, z) = (1, 1.5, 0) \quad \text{et} \quad f = -3.25.}$$

2.3.3 Optimisation : Résolution sous R

La production massive de données a entraîné une rupture non discutable avec les résolutions traditionnelles au profit des méthodes numériques. En outre, résoudre un problème d'optimisation à la main devient rapidement impraticable dès que le nombre de variables augmente. Déjà avec seulement quatre variables, la démarche manuelle devient laborieuse et sujette à des erreurs. Pour pallier cette limitation, les outils numériques nous offrent une solution élégante qui automatise les calculs tout en nous garantissant précision et rapidité. Ici naît le besoin crucial d'utiliser par exemple des outils tels que R qui s'impose alors comme un compagnon indispensable.

2.3.4 Modélisation du problème d'optimisation sous R

La résolution du problème d'optimisation quadratique sous contraintes linéaires nécessite une transcription sous R. Nous pouvons redéfinir la fonction objectif, une expression quadratique à trois variables, sous la forme matricielle :

$$f(x) = \frac{1}{2} x^\top Q x + c^\top x$$

où Q est une matrice symétrique 3×3 pour les termes quadratiques et c un vecteur pour les termes linéaires. Les contraintes sont exprimées sous la forme $A \cdot x \leq b$, où A représente les coefficients des inégalités et b leurs limites.

Pour modéliser le problème, nous utilisons le package `quadprog`, qui permet d'intégrer ces éléments et de résoudre le problème avec la fonction `solve.QP()`.

Pour être résolu par le package `quadprog`, le problème doit donc être reformulé de la manière suivante :

$$\min_x \frac{1}{2} x^\top Q x - d^\top x$$

où $d = -c$, ce qui adapte le terme linéaire $+c^\top x$ en $-d^\top x$ conformément aux spécifications du package. Les contraintes doivent également être réécrites sous la forme :

$$A^\top x \geq b$$

Pour utiliser `quadprog`, nous devons définir les matrices et vecteurs suivants :

— **Matrice** Q : correspond aux coefficients des termes quadratiques de la fonction objectif.

$$Q = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{bmatrix}$$

— **Vecteur** d : contient les coefficients des termes linéaires de la fonction objectif, avec des signes inversés pour correspondre à la formulation de `quadprog`.

$$d = \begin{bmatrix} 2 \\ 3 \\ -1 \end{bmatrix}$$

— **Matrice des contraintes** $A^\top$: chaque colonne représente une contrainte.

$$A^\top = \begin{bmatrix} -1 & 1 & 1 & 0 & 0 \\ -2 & 1 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

— **Vecteur des bornes** b_0 : contient les termes constants des contraintes.

$$b_0 = \begin{bmatrix} -4 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

2.3.5 Vérifications Préalables : Admissibilité et Convexité du Problème

Avant de résoudre le problème d'optimisation, nous allons vérifier certains points pour s'assurer que la résolution est possible et que les résultats auront du sens. 3 points nous sont principalement importants : Vérifier l'Existence d'une Solution (Région Admissible Non Vide), Vérifier si la Fonction Objectif est Convexe et enfin, vérifier la Faisabilité Numérique des Contraintes.

```r
###### Test de faisabilité ######
# Fonction de vérification du problème d'optimisation
verify_optimization_problem <- function(D, dvec, Amat, bvec) {
  # Test de convexité via les valeurs propres
  eigenvalues <- eigen(D)$values
  convexity_status <- all(eigenvalues > 0)

  # Affichage du résultat de convexité
  if (convexity_status) {
    cat("Test de convexité :\n")
    cat("- La fonction objectif est convexe (matrice D définie positive)\n")
    cat("- Valeurs propres :", eigenvalues, "\n\n")
  } else {
    cat("Test de convexité :\n")
    cat("- La fonction objectif n'est pas convexe\n")
    cat("- Valeurs propres problématiques :", eigenvalues[eigenvalues <= 0], "\n\n")
    return(FALSE)
  }

  # Test de faisabilité avec une matrice définie positive minimale
  D_test <- diag(1e-6, nrow(D))  # Matrice diagonale avec de petites valeurs positives
  dvec_test <- rep(0, length(dvec))

  # Tentative de résolution pour tester la faisabilité
  tryCatch({
    test_solution <- solve.QP(D_test, dvec_test, Amat, bvec, meq=0)
    if (!is.null(test_solution$solution)) {
      cat("Test de faisabilité :\n")
      cat("- Les contraintes sont faisables\n")
      cat("- Point admissible trouvé :", test_solution$solution, "\n\n")

      cat("Conclusion : Le problème est bien posé et peut être résolu.\n")
      return(TRUE)
    }
  }, error = function(e) {
    cat("Test de faisabilité :\n")
    cat("- Les contraintes sont infaisables\n")
    cat("- Erreur détectée :", e$message, "\n\n")

    cat("Conclusion : Le problème ne peut pas être résolu.\n")
    return(FALSE)
  })
}

# Utilisation de la fonction avec nos matrices définies précédemment
problem_status <- verify_optimization_problem(D, dvec, Amat, bvec)
```

Commentaire du code :

Pour le **Test de convexité**, ce code analyse les valeurs propres de la matrice $\mathbf{D}(\mathbf{Q})$ pour s'assurer qu'elle est définie positive, **condition nécessaire pour que la fonction objectif soit convexe et garantisse un minimum global**. Si $\mathbf{D}$ n'est pas définie positive, le test échoue et le problème est déclaré mal posé.

Pour le **Test de faisabilité des contraintes**, Une tentative de résolution simplifiée est effectuée avec une matrice $\mathbf{D}(\mathbf{Q})$ minimale et un vecteur **dvec** nul. Cela permet de vérifier si un point admissible existe qui satisfait les contraintes. En cas d'échec, le problème est déclaré infaisable.

Résultat de l'analyse du problème d'optimisation

```
> problem_status <- verify_optimization_problem(Q, dvec, Amat, bvec)

Test de convexité :
- La fonction objectif est convexe (matrice Q définie positive)
- Valeurs propres : 2 2 2

Test de faisabilité :
- Les contraintes sont faisables
- Point admissible trouvé : 0.5 0.5 0

Conclusion : Le problème est bien posé et peut être résolu.
```

Le résultat des deux tests nous permet de garantir que le problème d'optimisation est bien posé et peut être résolu efficacement.

2.3.6 Analyse du point optimal : $P = (1, \frac{3}{2}, 0)$

Résultats structurés

Solution optimale :

- $x = 1$
- $y = 1.5$
- $z = 5.551115 \times 10^{-17}$

Valeur optimale de la fonction objectif : -3.25

La résolution du problème d'optimisation a donné la solution optimale suivante :

- $x = 1, y = 3/2$ $z = 0$ (approximé à 5.551×10^{-17}, considéré comme nul en pratique).
- La valeur minimale de la fonction objectif est -3.25.

Vérification des Contraintes :
Les contraintes sont satisfaites :

- $x + 2y + z = (1) + 2 \times (3/2) + (0) = 4 \leq 4$,
- $x + y = (1) + (1) = 2 \geq 1$,
- $x, y, z \geq 0$.

La solution est cohérente et respecte toutes les contraintes. Le point optimal trouvé $(x = 1, y = 1.5, z = 0)$ minimise la fonction objectif tout en restant dans la région admissible définie par les contraintes.

2.3.7 La Validité du point optimal

Même si notre point optimal est cohérent et respecte toutes les contraintes, il est tout aussi important de vérifier la validité de ce dernier.

Output R

```
# Application de la vérification
> check_KKT(c(1.00, 1.50, 0.00), NULL)
[1] "Gradient de la fonction objectif:"
[1] 0 0 1
[1] "Vérification des contraintes:"
[1] "Contrainte 1: TRUE"
[1] "Contrainte 2: TRUE"
[1] "Non-négativité: TRUE"

# Application de la vérification
> check_perturbation(c(1.00, 1.50, 0.00))
[1] "Le point semble être un minimum local"
[1] TRUE
```

La vérification des conditions KKT (Karush-Kuhn-Tucker) nous permet de dire que notre point optimal trouvé (1.00, 1.50, 0.00) est cohérent avec les conditions d'optimalité et satisfait à la fois les contraintes du problème et les conditions nécessaires d'optimalité locale. **Notre point est un minimum local.** Par ailleurs, **il est aussi global** parce que notre fonction objectif est strictement convexe, ce qui garantit qu'un minimum local est nécessairement un minimum global.

2.3.8 Visualisation du Point Optimal

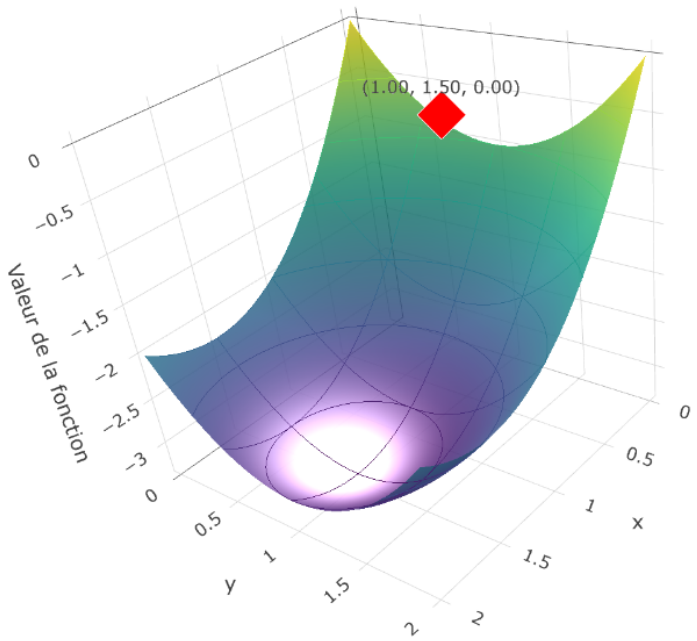


FIGURE 14 – Visualisation de l'optimisation quadratique

La visualisation illustre la surface de la fonction objectif $f(x, y, z)$, avec une forme convexe qui garantit l'existence d'un minimum global. Le point $(1.00, 1.50, 0.00)$, marqué en rouge, est situé au sommet inférieur de la région admissible, ce qui correspond au résultat optimal obtenu par

calcul. La région autour du point optimal présente une pente douce, confirmant la minimisation effective dans un domaine faisable. Les projections montrent que $z = 0$, ce qui réduit l'influence de z sur la solution optimale. Cette représentation graphique renforce la cohérence des calculs et offre une interprétation visuelle claire des résultats.

2.4 Partie 4 : SOUS-GRAPHE RECOUVRANT OPTIMAL DU METRO PARISIEN

Toujours dans cette logique d'optimisation, imaginons que cette fois, on soit appelés à concevoir le réseau de transport idéal pour une ville complexe comme Paris. La question qu'on se pose : Comment connecter toutes les stations de métro de manière optimale, en **minimisant** les coûts d'installation et en sélectionnant uniquement les trajets nécessaires pour garantir un réseau à la fois économique, cohérent et sans redondance ? Ainsi, dans cette partie du projet, notre objectif est de trouver **un sous-graphe recouvrant optimal d'un graphe connexe valué en utilisant l'algorithme de KRUSKAL**.

Les données que nous allons utiliser dans cette résolution proviennent d'une matrice de distances pondérées, qui intègre à la fois la proximité entre les stations et des pénalités spécifiques pour les transitions entre lignes distinctes, représentées par une pondération exponentielle.

2.4.1 Définitions des concepts utilisés

Le Graphe Valué ou Pondéré :

Un graphe pondéré est une structure mathématique $G = (V, E, w)$, où V représente l'ensemble des sommets (ou nœuds), E l'ensemble des arêtes (ou connexions entre sommets), et $w : E \to \mathbb{R}^+$ une fonction de poids associant une valeur numérique (appelée poids) à chaque arête. Dans le contexte du métro parisien, les sommets représentent les stations, les arêtes les connexions entre elles, et les poids traduisent les distances ou pénalités associées aux trajets.

Le Minimum Spanning Tree (MST) :

Un **arbre couvrant minimal** (MST) est un sous-ensemble d'arêtes connectant tous les sommets d'un graphe sans former de cycles, tout en minimisant la somme totale des poids. Mathématiquement, soit $G = (V, E, w)$ un graphe pondéré connexe. Le MST est défini comme :

$$T^* = \arg\min_{T \subseteq E} \sum_{e \in T} w(e)$$

sous la contrainte que T forme un arbre couvrant. Dans ce projet, le MST représente une version optimisée du réseau de métro, minimisant les distances ou pénalités tout en maintenant la connectivité entre les stations.

Comment fonctionne : l'Algorithme de Kruskal ?

L'algorithme de Kruskal ou encore le **greedy algorithm** nous permet de trouver le MST de notre graphe pondéré. Il classe d'abord les arêtes par ordre croissant de poids, puis les ajoute au sous-graphe en évitant la formation de cycles. Il permet donc la détermination d'un arbre (un graphe connexe sans cycles) couvrant minimal (T^*) pour un graphe pondéré connexe $G = (V, E, w)$, où V est l'ensemble des sommets, E l'ensemble des arêtes, et $w : E \to \mathbb{R}^+$ une fonction de poids. Il procède comme suit :

1. **Tri des arêtes** : Les arêtes de E sont triées par ordre croissant de poids $w(e)$, produisant une séquence $(e_1, e_2, \ldots, e_m)$, où $w(e_i) \leq w(e_{i+1})$.
2. **Construction de** T^* : Initialement, $T^* = \emptyset$. Pour chaque arête $e_i \in E$, on ajoute e_i à T^* si et seulement si $T^* \cup \{e_i\}$ ne contient pas de cycle. Autrement dit, T^* reste un sous-graphe acyclique connexe couvrant.
3. **Condition d'arrêt** : L'algorithme s'arrête lorsque $|T^*| = |V| - 1$, où $|T^*|$ est le nombre d'arêtes dans T^*.

La Matrice de distance pondérée :

La matrice de distance pondérée M $(n \times n)$ associe un poids $M[i, j]$ à chaque paire de sommets i et j. Dans cette partie du projet :

— Si deux stations appartiennent à la même ligne, $M[i, j]$ représente le nombre de stations entre elles.

— Si elles appartiennent à des lignes différentes, une pénalité est ajoutée sous forme d'une pondération exponentielle $M[i, j] = \exp(10)$, ce qui reflète le coût additionnel des correspondances.

2.4.2 Préparation et vérification des données du graph

La vérification des données est en effet essentielle pour garantir la fiabilité de nos résultats. Identifier les valeurs manquantes (NA) nous permet d'éviter des erreurs dans la modélisation en les traitant correctement. Par ailleurs, vérifier la symétrie de la matrice assure la cohérence des distances dans un graphe **non orienté**, tandis que l'analyse des valeurs uniques aide à détecter d'éventuelles anomalies dans les pondérations. Enfin, la diagonale doit être nulle ($M[i, i] = 0$) pour refléter l'absence de distance entre une station et elle-même. Ces étapes sont indispensables pour éviter des incohérences qui pourraient compromettre la construction de notre graphe et la détermination du Minimum Spanning Tree (MST).

Section de vérification

```
> sum(is.na(metro_matrix))
[1] 0
> is_symmetric <- all(metro_matrix == t(metro_matrix))
> print(paste("La matrice est-elle symétrique ?", is_symmetric))
[1] "La matrice est-elle symétrique ? TRUE"
> unique_values <- sort(unique(as.vector(as.matrix(metro_matrix))))
> print("Valeurs uniques dans la matrice :")
[1] "Valeurs uniques dans la matrice :"
> print(unique_values)
 [1]  0.00  1.00  2.00  3.00  4.00  5.00  6.00  7.00  8.00  9.00
[11] 10.00 11.00 12.00 13.00 14.00 15.00 16.00 17.00 18.00 19.00
[21] 20.00 21.00 22.00 23.00 24.00 25.00 26.00 27.00 28.00 29.00
[31] 30.00 31.00 32.00 33.00 34.00 35.00 36.00 37.00 38.00 22026.47
> diag_values <- diag(as.matrix(metro_matrix))
> print("Vérification des valeurs de la diagonale :")
[1] "Vérification des valeurs de la diagonale :"
> print(table(diag_values))
diag_values
  0 320
> # Identification des composantes connexes
> components <- components(temp_graph)
> print(paste("Nombre de composantes connexes :", components$no))
[1] "Nombre de composantes connexes : 1"
```

Les résultats nous confirment que **la matrice de distances est bien structurée et prête pour la modélisation**. L'absence de valeurs manquantes (NA) et la symétrie parfaite de la matrice nous garantissent la cohérence des données pour un graphe non orienté. Les valeurs uniques identifiées, incluant la pondération exponentielle (exp(10)) pour les correspondances inter-lignes, nous montrent donc une pondération logique et cohérente. De plus, la diagonale nulle ($M[i, i] = 0$) reflète correctement l'absence de distance entre une station et elle-même. Par ailleurs, le fait d'avoir une seule composante connexe signifie que **notre réseau est entièrement connecté et qu'il existe un chemin entre n'importe quelle paire de stations**. En effet, cela nous assure une base fiable pour la construction du graphe et l'application rigoureuse de l'algorithme de Kruskal.

2.4.3 Visualisation du graphe pondéré complet

Avant de passer à la construction de notre arbre recouvrant minimal (ce qui nous est demandé), dressons un portrait global du réseau métropolitain parisien afin de voir l'ensemble des stations et des liaisons telles qu'elles apparaissent dans notre matrice de distances. En visualisant chaque station (comme un nœud) et chacune de ses liaisons (sous forme d'arête pondérée), nous pouvons intuitivement constater la densité du réseau, appréhender la hiérarchie des poids (faible entre stations voisines, élevé pour les correspondances inter-lignes), et ainsi repérer rapidement des stations centrales ou des terminus en périphérie. Cela pourrait nous permettre en effet de voir toute la complexité du réseau, notamment le nombre de chemins potentiels reliant deux stations.

Pour ce faire, nous utilisons la matrice des distances, dont chaque entrée (i, j) détermine le

poids de la liaison entre les stations i et j. Ensuite, **nous convertissons ces informations en un graphe non orienté, pondéré, afin d'attribuer un poids à chaque arête**. Grâce à la librairie `ggraph`, nous plaçons les stations selon un algorithme de disposition et colorons les arêtes selon leur poids (palette "viridis").

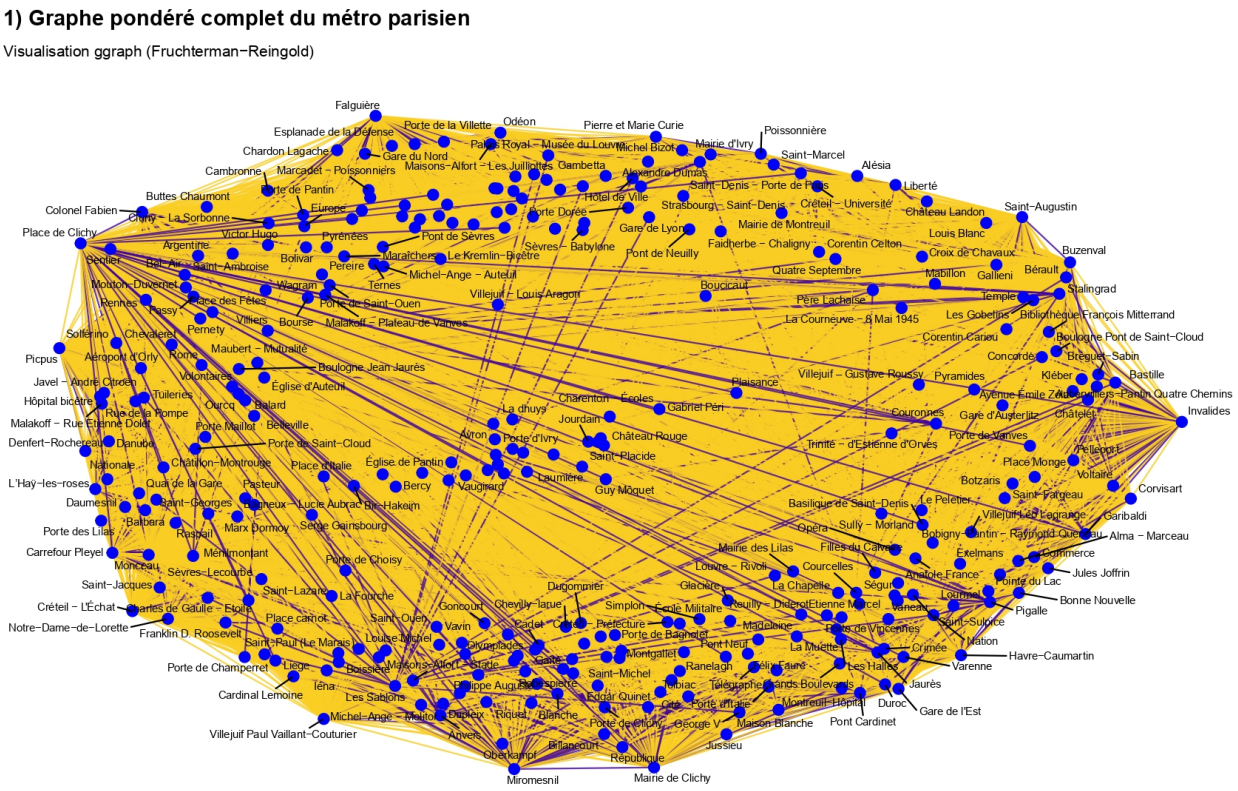


FIGURE 15 – Réseau complet pondéré du métro parisien

À première vue, nous n'observons aucune station isolée. Les multiples correspondances suggèrent qu'il existe divers chemins (parfois redondants) pour relier deux points éloignés. Les liaisons intra-ligne se distinguent nettement des grandes correspondances inter-lignes, souvent associées à des poids exponentiels. Visuellement, cela se traduit par des arêtes plus ou moins colorées ou denses.

2.4.4 Construction et analyse du MST

Après avoir visualisé l'ensemble de notre réseau, nous avons constaté qu'il contenait de nombreuses arêtes potentiellement redondantes. L'étape suivante consisterait donc à construire notre sous-graphe recouvrant Minimal (le MST), puis à l'analyser pour mieux comprendre quelles stations se révèlent essentielles à la connectivité globale.

Dans un réseau de transport comme le métro, il existe souvent plusieurs chemins reliant deux stations. Cependant, lorsque nous voulons étudier la structure minimale qui maintient la connectivité, **le MST devient un outil pertinent pour éviter les cycles inutiles, réduire le nombre d'arêtes au strict nécessaire.**

Pour ce faire, nous avons utilisé la fonction `mst()` de la librairie `igraph`, qui implémente en interne l'algorithme de Kruskal permettant de trouver le sous-graphe recouvrant minimal pour le graphe pondéré. Le MST est directement calculé à partir du graphe pondéré complet, où chaque station est un nœud et chaque liaison est caractérisée par un poids (nombre de stations intermédiaires ou exp(10) en cas de correspondance).

Par ailleurs, pour valider notre MST, nous avons bien vérifié que le nombre d'arêtes obtenues est de $N-1$, avec N le nombre total de stations et nous avons testé la **connectivité** du sous-graphe résultant, qui est resté connecté malgré la négligence d'arêtes inutiles.

Output R

```
> print(paste("Nombre de nœuds MST :", num_nodes))
[1] "Nombre de nœuds MST : 320"
> print(paste("Nombre d'arêtes MST :", num_edges))
[1] "Nombre d'arêtes MST : 319"

> if (num_edges == num_nodes - 1) {
+   print("MST valide : connexe et acyclique.")
+ } else {
+   print("Le MST peut contenir des cycles ou être incomplet.")
+ }
[1] "MST valide : connexe et acyclique."

> if (is.connected(mst_graph)) {
+   print("Le MST est connecté.")
+ } else {
+   print("Le MST n'est pas connecté.")
+ }
[1] "Le MST est connecté."
```

En appliquant l'algorithme de Kruskal, nous avons obtenu un MST qui conserve toutes les stations (chacune correspond à un nœud), qui n'emploie qu'un minimum d'arêtes pour maintenir la connectivité et qui néglige toutes les redondances éventuelles (aucun cycle inutile).

FIGURE 16 – MST du métro parisien

2.4.5 Que dire de la structure actuelle du métro Parisien ?

Visuellement, le MST affiche beaucoup moins d'arêtes que le graphe initial, ce qui reflète l'idée **qu'un chemin unique suffit à relier deux stations dans un sous-réseau acyclique.** En effet, cette construction nous permet d'identifier les liaisons clés **sans lesquelles la connectivité serait rompue**. Nous pouvons ainsi discerner les stations importantes grâce à des mesures de centralité.

2.4.6 Analyse de centralité : La Défense et Jussieu dans le réseau métropolitain

Notre objectif dans cette section, c'est de voir et de discuter de l'importance relative de ces deux stations dans le réseau métropolitain parisien. Cela revient donc à estimer l'importance structurelle de ces stations. **Pour évaluer l'impact que chaque station a sur la connectivité du réseau, nous avons quantifié leur absence sélective dans le réseau.** Ainsi, la question se pose, que devient le réseau sans Jussieu ? ou encore que devient-il sans la Défense ? Ces questions peuvent paraître simplistes mais y répondre exigerait une méthodologie conforme.

Par ailleurs, pour répondre à cette question, nous avons utilisé deux principales méthodes : **la centralité d'intermédiarité (Betweenness Centrality) et la centralité de proximité (Closeness Centrality).**

Le premier, par exemple, mesure la fréquence à laquelle un nœud (station) se trouve sur les plus courts chemins entre toutes les paires de nœuds du réseau. Pour ce faire, on calcule pour chaque

paire de nœuds, la proportion des plus courts chemins qui passent par le nœud étudié(dans notre cas, on le fera pour Jussieu et pour la Défense). Une valeur élevée indique par exemple que la station est un "pont" important dans le réseau. Ainsi, **plus la valeur est élevée, plus la station est critique pour la connectivité globale**. Par conséquent, cela nous permettra de déterminer laquelle des deux stations (Jussieu ou La Défense) est la plus cruciale en termes de connexions et dans notre réseau.

De sorte à compléter notre analyse, le deuxième est une méthode complémentaire qui nous permet de mesurer la proximité moyenne d'un nœud à tous les autres nœuds du réseau. Il fait l'inverse de la somme des plus courts chemins entre le nœud étudié et tous les autres nœuds afin d'avoir des outputs intuitives (parce qu'on veut qu'une station plus centrale ait un score plus élevé). Une valeur élevée nous indique que la station est facilement accessible depuis les autres stations. Ainsi, **plus la valeur est élevée, plus la station est centrale**. Par conséquent, cela nous permettra d'évaluer laquelle de ces deux stations est la plus accessible et la mieux positionnée dans l'ensemble du réseau.

De manière résumé, la Betweenness nous dit quelle station est la plus importante en terme de flux et de connectivité et la Closeness nous indique quelle station est la plus centrale en terme d'accessibilité.

Avant de discuter de leur importance relative par les deux méthodes, visualisons le MST sans chacun d'elles.

2.4.7 Le MST sans Défense et le MST sans Jussieu

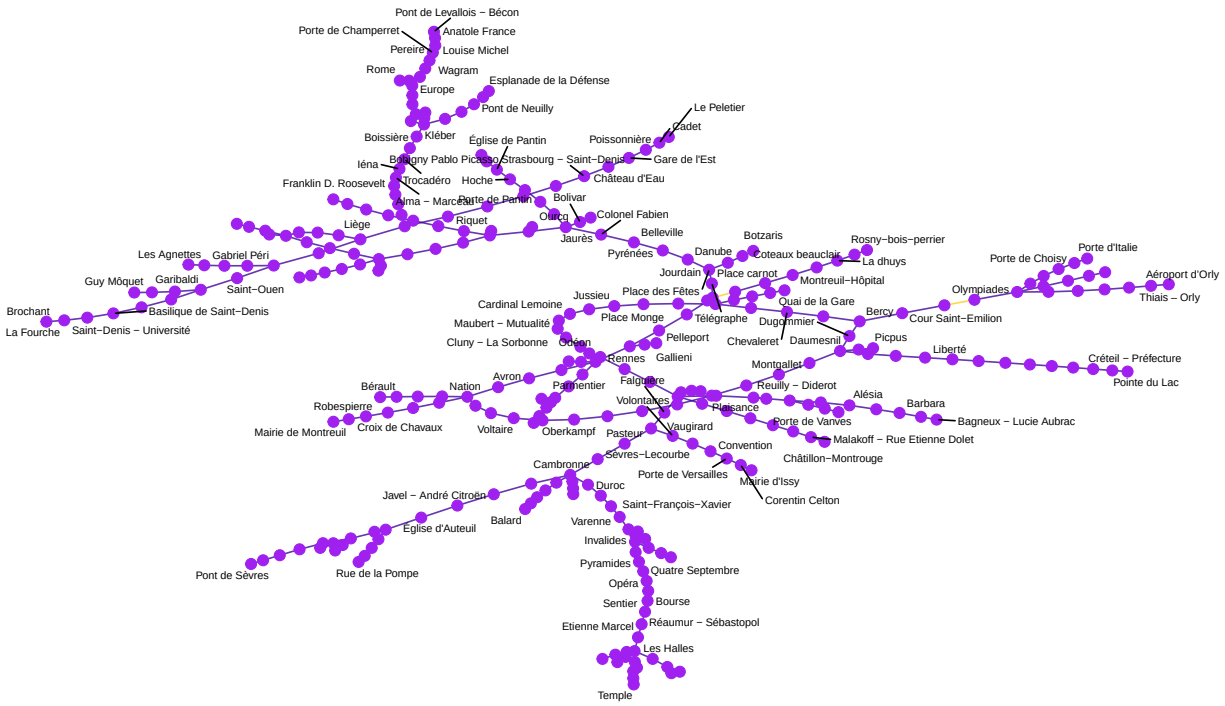


FIGURE 17 – MST du métro parisien sans Défense

3) MST du métro SANS 'Jussieu'

Impact de la suppression de Jussieu

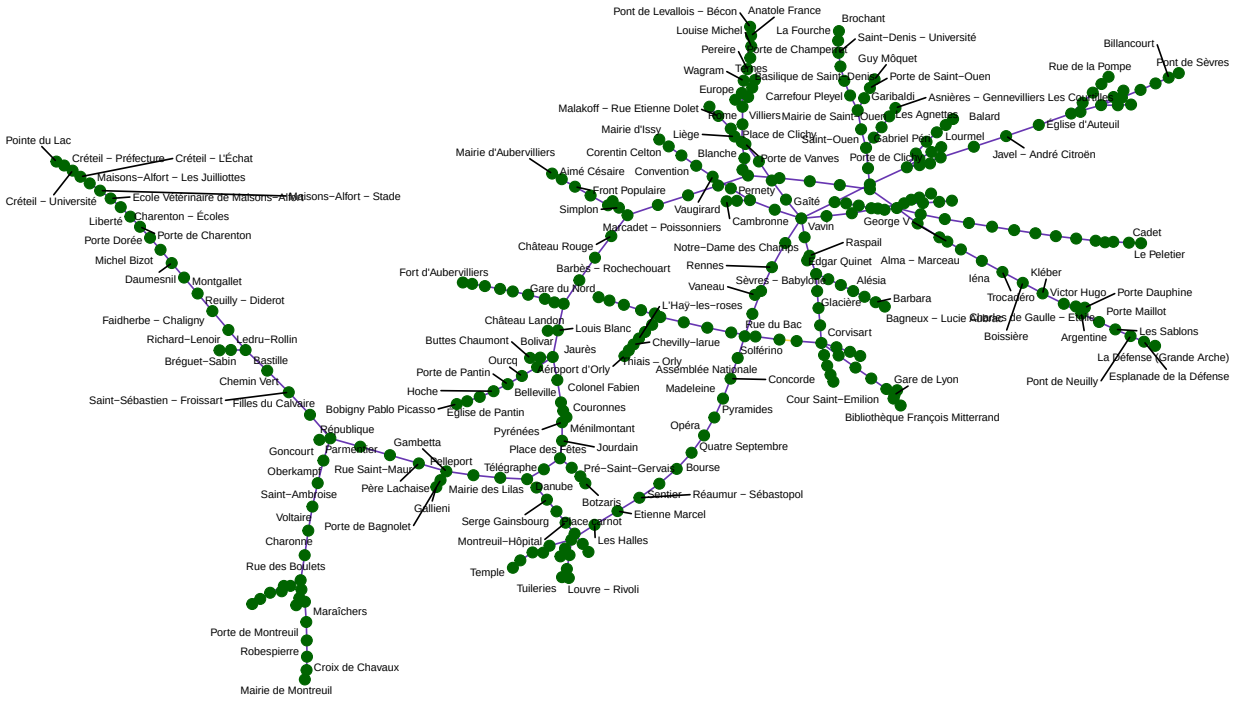


FIGURE 18 – MST du métro parisien sans Jussieu

2.4.8 Importance Relative : La Betweenness

À partir de la fonction `Betweenness(mst_graph, normalized = TRUE)`, nous avons obtenu pour chaque station un score d'intermédiarité. Nous avons ensuite extrait et comparé les valeurs pour « La Défense (Grande Arche) » et « Jussieu ».

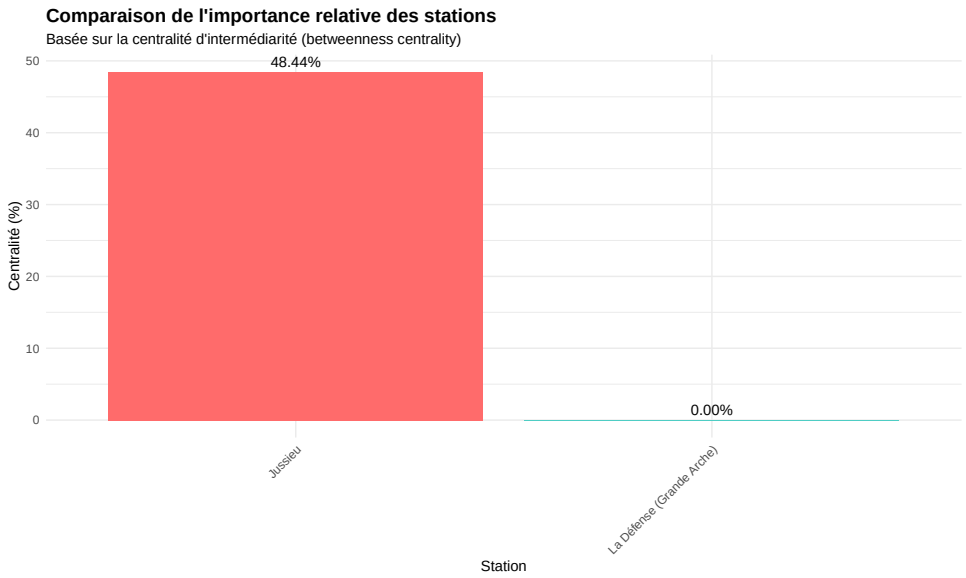


FIGURE 19 – Importance Basée sur la Centralité d'Intermédiarité

2.4.9 Interprétation et Conclusion :

Le score de Betweenness de 0,48 pour Jussieu indique que cette station se retrouve plus souvent sur les chemins minimaux reliant d'autres stations et qu'elle sert de point de passage pour environ 48% des plus courts chemins possibles dans le réseau. Cela signifie qu'en l'absence de Jussieu, de nombreux itinéraires deviendraient plus longs ou moins directs. Une centralité de 0 pour La Défense signifie qu'elle ne sert pas de point de passage pour les plus courts chemins entre d'autres stations dans le MST. Cela suggère qu'elle est probablement située à une extrémité du réseau.

Par conséquent, en termes de centralité d'intermédiarité, Jussieu est relativement plus importante que La Défense.

2.4.10 Importance Relative : La closeness

En complément, nous avons ensuite évalué la centralité de proximité via la fonction `closeness(mst_graph, normalized = TRUE)`. De la même façon, nous avons isolé les scores de « La Défense (Grande Arche) » et « Jussieu » pour une comparaison directe.

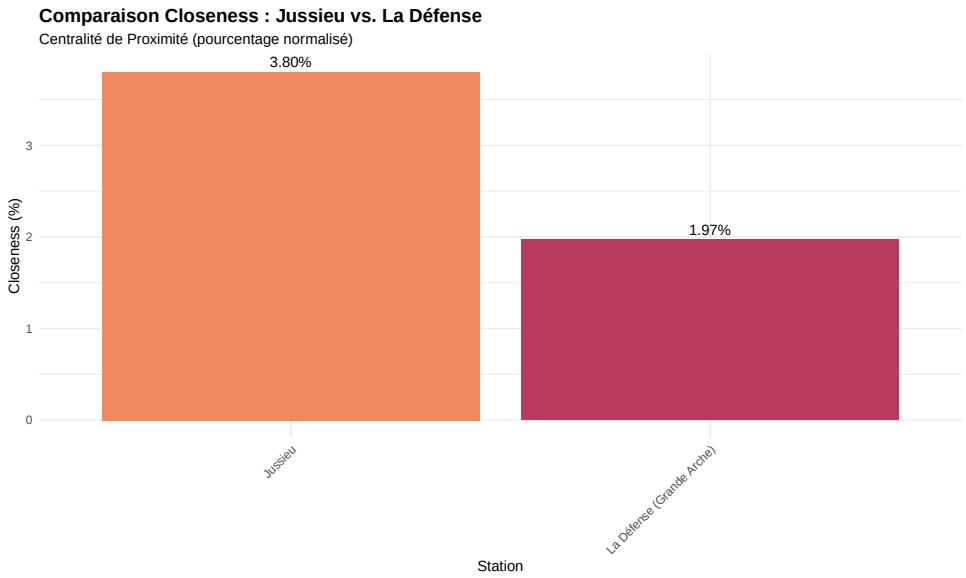


FIGURE 20 – Importance Basée sur la Centralité de proximité

En ce qui concerne la Closeness, Jussieu présente également un score plus important (0,038 contre 0,0197). Autrement dit, Jussieu est, en moyenne, «plus proche» du reste du MST que La Défense ne l'est : **il est plus rapide (en termes d'arêtes) de joindre Jussieu depuis d'autres stations que d'atteindre La Défense.**

Par conséquent, en termes de centralité de proximité, Jussieu est plus centrale que la Défense et donc plus efficace en termes de temps de parcours vers les autres stations.

2.4.11 Conclusion Finale :

Les deux mesures (betweenness et closeness) concordent parfaitement. Jussieu apparaît comme une station plus critique (importante) que La Défense car elle est plus centrale (meilleure closeness) et sert davantage de point de passage (meilleure betweenness).

Jussieu joue donc un rôle plus stratégique dans la structure globale du réseau de métro parisien.

Par ailleurs, ces résultats sont cohérents avec la réalité géographique puisque Jussieu est située dans Paris intra-muros, tandis que La Défense est en périphérie, ce qui explique naturellement ces différences de centralité.