Université du Québec à Montréal (UQAM)
Faculté des sciences

ACT3035– Examen Intra (R)
Laboratoire d'actuariat

Enseignant: Noureddine Meraihi

2020/06/30

Nom: _____

Code permanent: _____

Signature: _____

Cet examen contient 8 pages (incluant la page couverture) et 4 questions sur un total de 45 points. Bon succès à tous!

Distribution des points

Question	Points	Score
1	6	
2	14	
3	10	
4	15	
Total:	45	

Instructions

- L'examen commence à 17:30 pour une durée de 180 minutes;
- Vous avez le droit d'utiliser votre ordinateur **SEULEMENT** pour;
  - Vous connecter à *Zoom Meetings*
  - consulter le questionnaire de l'examen
  - Écrire vos réponses sur le cahier de réponse BRUH123456.R
- Il est strictement **interdit** d'utiliser un quelconque moyen de communication pendant l'examen;
- Il est strictement **interdit** de faire des recherches sur le web;
- Vous avez le droit de consulter vos notes de cours personnelles;
- Vous avez le droit de consulter les notes de cours du livre nour.me/act3035book;
- Vous avez le droit de consulter tout le matériel du cours se trouvant dans mon github: https://github.com/nmeraihi/ACT3035
- Vous avez le droit de consulter l'aide de RStudio;
- Il est strictement interdit de faire des recherches sur le web;
- Pour toutes les questions, le terme **df** désigne *data frame*
- N'oubliez pas de sauvegarder aussi souvent que possible (Ctrl+s)!
- Le nom de votre fichier de réponse doit contenir votre code permanent MERN12345678.R
- L'examen compte pour 50% de la note finale;
- Vous serez informés par courriel/Slack lorsque l'examen sera corrigé.

1. Supposons que nous avons des données générées à partir d'une distribution gaussienne dont la densité de probabilité est donnée par:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \mathrm{e}^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \tag{1}$$

   (a) (2 points) Générer un vecteur X de taille $n = 100$ ainsi qu'un vecteur ϵ de taille $n = 100$ tous les deux tirés de la même distribution présentée à l'équation (1). Où votre μ et σ sont les valeurs par défaut, soit $(0, 1)$.

```
set.seed(3035)
X =rnorm(100)
eps = rnorm(100)
```

   (b) (4 points) Générer un vecteur Y de taille $n = 100$ selon le modèle suivant;

$$Y = \beta_0 + \beta_1 X + \beta_2 X^2 + \beta_3 X^3 + \epsilon, \tag{2}$$

   où $\beta_0, \beta_1, \beta_2$ ainsi que β_3 sont des constantes de votre choix.

```
beta0 = 3
beta1 = 2
beta2 = -3
beta3 = 0.3
Y = beta0 + beta1 * X + beta2 * X^2 + beta3 * X^3 + eps
```

2. L'épidémie de maladie Coronavirus de 2019-2020 est une épidémie ayant commencé au début du mois de décembre 2019 dans la ville de Wuhan, en Chine centrale, avant de se propager dans le monde.

   Puisque nous avons des données très à jour à propos de l'évolution de cette épidémie, nous allons nous "amuser" dans cet examen à ressortir les faits saillants sur ces données.

   (a) (1 point) À partir des données [1] dans le fichier `covid_19_data.csv`, créer un df appelé `df_corona`

```
df_corona<-read.csv("intra/novel-corona-virus-2019-dataset//covid_19_data.
    csv")
```

   (b) (1 point) À partir des données [1] dans le fichier `covid_19_confirmed.csv`, créer un df appelé `df_location`

```
df_location<-read.csv("intra/novel-corona-virus-2019-dataset//time_series_
    covid_19_confirmed.csv") %>% select(1:4)
```

[1] Attention: Assurez-vous de faire référence au lien GitHub ou le répertoire relatif de la trousse d'examen

(c) (1 point) Modifier votre `df_location` afin que celui contiennes seulement les informations présentés dans le tableau (c) où les variables `lat` et `long` sont la moyennes de la latitude et longitude pour chaque pays

Country.Region	lat	long
Australia	-33.52018	146.94955
Belgium	50.50390	4.46990
Cambodia	12.56570	104.99100
Canada	45.30693	-94.58317
Egypt	26.82060	30.80250
Finland	61.92410	25.74820

```
df_corona <- df_corona %>%
left_join(df_location)


df_location_country <- df_location %>%
  group_by(Country.Region) %>%
  summarise(lat = mean(Lat),
            long = mean(Long))
```

(d) (1 point) Supprimer toute observation contenant une valeur `na` dans ce nouveau df `df_corona_loc`

```
df_corona <- df_corona %>%
    na.omit()
```

(e) (1 point) Utiliser la fonction `as.Date` afin de créer une nouvelle variable appelée `date` qui est simplement l'information contenue dans la variable `ObservationDate` sous le format **'%m/%d/%y'**

```
df_corona$date <- as.Date(df_corona$ObservationDate, format = '%m/%d/%y')
```

(f) (5 points) Créer un tableau qui vous indique le nombre de cas (*Confirmed, Deaths, Recovered*) par pays. Ajoutez une nouvelle variable qui vous donne le taux de décès calculé comme suit:

$$\text{death\_rate} = \frac{\text{max\_deaths}}{\text{max\_confirmed} + \text{max\_recovered} + \text{max\_deaths}} \tag{3}$$

```
df_stat_by_country <- df_corona_country %>%
  filter(Country.Region != 'Others') %>%
  group_by(Country.Region) %>%
  summarise(cum_confirmed = max(Confirmed),
            cum_deaths = max(Deaths),
            cum_recovered = max(Recovered)) %>%
  mutate(death_rate = round(cum_deaths/(cum_confirmed + cum_recovered + cum_
      deaths), 3)) %>%
  left_join(df_corona_country %>%
              select(Country.Region, Lat, Long)) %>%
```

```
  unique() %>%
  ungroup()
```

Country.Region	max_confirmed	max_deaths	max_recovered	death_rate	lat	long
Australia	27	0	11	0.000	-33.52018	146.94955
Belgium	8	0	1	0.000	50.50390	4.46990
Cambodia	1	0	1	0.000	12.56570	104.99100
Canada	26	0	6	0.000	45.30693	-94.58317
Egypt	2	0	1	0.000	26.82060	30.80250
Finland	6	0	1	0.000	61.92410	25.74820
France	191	3	12	0.015	46.22760	2.21370
Germany	159	0	16	0.000	51.16570	10.45150
Hong Kong	100	2	36	0.014	22.31930	114.16940
India	5	0	3	0.000	20.59370	78.96290
Iran	1501	66	291	0.036	32.42790	53.68800
Italy	2036	52	149	0.023	41.87190	12.56740

(g) (4 points) Créer un graphique qui vous montre le taux de décès par pays. Afin d'afficher les pays ayant le plus grand taux de décès en ordre décroissant, comme illustré à la figure (1), vous pouvez utiliser la fonction `reorder`.

```
country_death_rate_plot <- df_stat_by_country %>%
  arrange(-death_rate) %>%
  head(10) %>%
  ggplot(aes(reorder(Country.Region, death_rate), death_rate)) +
  geom_bar(stat = 'identity', fill = 'red', colour = 'red', alpha = 0.75,
      size = 1) +
  scale_y_continuous() +
  labs(x = '', y = '',
       title = 'Taux de décès') +
  coord_flip()
```

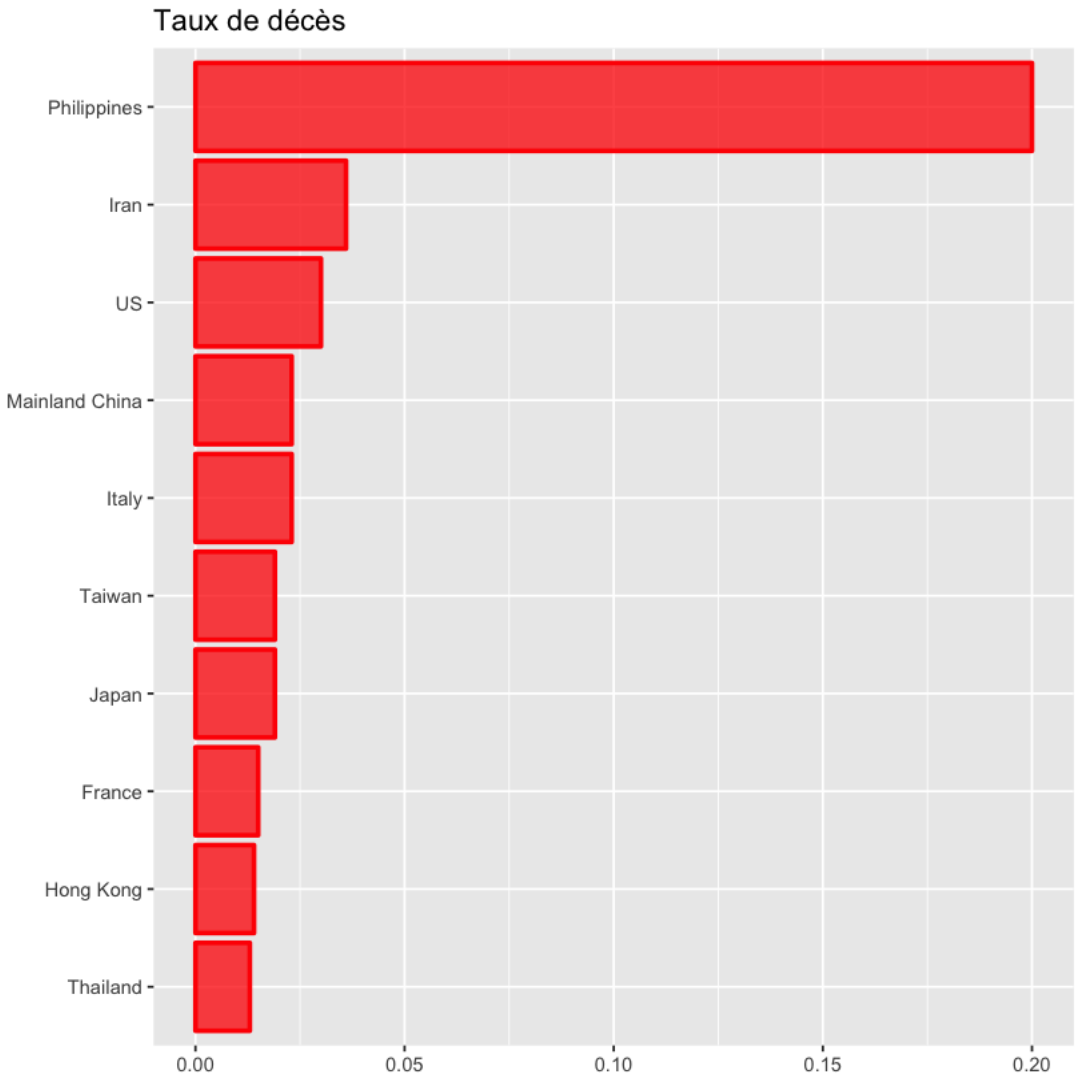

Figure 1: Taux de décès des personnes touchés par le virus Covid-19

3. Dans cette question, nous considérons un ensemble de données constitué des prix des logements dans une région aux États-Unis, qui comprend les logements vendus entre mai 2014 et mai 2015. L'ensemble de données comporte 21613 observations et 21 variables. Toutefois, pour des raisons de simplicité, nous sélectionnons la variable réponse **prix** (`Price`), et le prédicteurs **nombre de chmabre** (`bedrooms`) et **l'aire habitable** du logement (`sqft_living`).

   (a) (5 points) Ajustez un modèle de régression linéaire simple ayant comme variable réponse `Price` et les deux variables explicatives `sqft_living` et `bedrooms`.

```r
kc_house_data<-read.csv("intra//kc_house_data.csv")
model <- lm(price ~sqft_living + bedrooms, data = kc_house_data)
summary(model)$r.squared
```

   (b) (5 points) En statistique, le coefficient de détermination, noté R^2 ou r^2, est une mesure de la qualité de la prédiction d'une régression linéaire.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{4}$$

où n est le nombre d'observations, y_i la valeur de l'observation i, $\hat{y}_i$ la valeur prédite correspondante et $\bar{y}$ la moyenne des observations.

Créez une fonction appelée `Rsquared` qui vous permet de calculer le coefficient de détermination. Vous pouvez vérifier si vous avez la bonne réponse en comparant le résultat de votre fonction avec celle générée par le sommaire de votre modèle ajusté dans la question a).

```
Rsquared <- function(y, yhat) {
    rss <- sum((y - yhat)**2)
    tss <- sum(( y - mean(y))**2 )
    return( 1 - rss / tss )}

Rsquared(kc_house_data$price, model$fitted.values)
```

4. La table freTH0002 (resp. freTF0002) a été établie à partir des observations de l'Institut national de la statistique et des études économiques (INSEE) collectées sur la population masculine française (resp. la population féminine française).

   (a) (5 points) Vous savez tous que la probabilité qu'une personne âgée d'exactement un certain âge x survivent encore t années, c'est-à-dire qu'elle vit au moins jusqu'à l'âge $t + x$ est donnée par:

   $$ {}_tp_x = \frac{\ell_{x+t}}{\ell_x} \tag{5}$$

   Créez une fonction appelée `tpx(t,x)` qui vous permet de calculer la probabilité ${}_tp_x$ définie à l'équation (5).

   Afin de vérifier votre réponse, si l'on tape `tpx(10,30)` pour la table des femmes, on obtient: 0.9931359

   (b) (3 points) Supposons maintenant que l'on vous fournit la table de probabilité de décès `mortalityTable.csv` (car, gentiment, je crois que pour plusieurs, ce sera un peu difficile de la construire dans le cadre d'un examen), quelle est donc la probabilité qu'une personne âgée de 23 ans survive les 38 prochaines années.

   Écrivez seulement le code R qui vous donne la réponse tirée table de probabilité de décès. Remarquez que vous pouvez vérifier votre réponse avec la fonction `tpx(t,x)`.

   (c) (7 points) Nous savons que la moyenne de la durée de vie future K_x d'un produit est donnée par $\mathbb{E}[K_x] = \sum_{k=0}^{\infty} k\,({}_kp_x - {}_{k+1}p_x)$. En assurance-vie, les actuaires désignent cette moyenne par e_x et l'appellent l'espérance de vie. On peut facilement montrer que cette espérance de vie se réduit à l'expression suivante;

   $$e_x = \sum_{k=1}^{\infty} {}_kp_x \tag{6}$$

En utilisant la table de probabilité de décès `mortalityTable.csv`, créez une fonction appelée `esperance_x(x)` qui vous permet de calculer cette espérance. Par exemple, pour une femme âgée de 23 ans, nous obtenons 60.1389.

Fin de l'examen

- N'oubliez pas d'identifier votre fichier `BRUH123456.R` que vous remettez avec votre code permanent comme nom du fichier.
- Déposez votre cahier de réponse MERN12345678.R cliquant sur le lien suivant: **https://bit.ly/3gb1t1Q**.
- Vous serez avisé par courriel quand les notes seront disponibles.