Homework 4: Diffusion of Tetracycline

We continue examining the diffusion of tetracycline among doctors in Illinois in the early 1950s, building on our work in lab 6. You will need the data sets `ckm_nodes.csv` and `ckm_network.dat` from the labs.

1. Clean the data to eliminate doctors for whom we have no adoption-date information, as in the labs. Only use this cleaned data in the rest of the assignment.

```r
library(tidyverse)
ckm_nodes <- read_csv('data/ckm_nodes.csv')
noinfor <- which(is.na(ckm_nodes$adoption_date))
ckm_nodes <- ckm_nodes[-noinfor, ]
ckm_network <- read.table('data/ckm_network.dat')
ckm_network <- ckm_network[-noinfor, -noinfor]
```

2. Create a new data frame which records, for every doctor, for every month, whether that doctor began prescribing tetracycline that month, whether they had adopted tetracycline before that month, the number of their contacts who began prescribing strictly *before* that month, and the number of their contacts who began prescribing in that month or earlier. Explain why the dataframe should have 6 columns, and 2125 rows.

```r
# 获取基本信息
n_doctors <- nrow(ckm_nodes)  # 医生数量
months <- 1:17  # 月份范围：1 到 17 个月
n_months <- length(months)
```

```
# 创建空的数据框
doctor_month_data <- data.frame(
  doctor_id = rep(1:n_doctors, each = n_months),  # 医生 ID
  month = rep(months, n_doctors),                 # 月份
  started_this_month = NA,                        # 本月是否开始使用
  adopted_before = NA,                            # 本月之前是否已采用
  contacts_before = NA,                           # 本月之前联系人中的采用者数量
  contacts_up_to_now = NA                         # 本月及之前联系人中的采用者数量
)

# 为每个医生和每个月填充数据
for (i in 1:n_doctors) {
  doctor_adoption_month <- ckm_nodes$adoption_date[i]
  for (month in months) {
    row_idx <- (i - 1) * n_months + month
    # 本月是否开始使用四环素
    doctor_month_data$started_this_month[row_idx] <-
      (month == doctor_adoption_month)
    # 本月之前是否已经采用
    doctor_month_data$adopted_before[row_idx] <- (doctor_adoption_month < month)
    # 计算联系人中的采用情况
    # 获取当前医生的联系人 (邻接矩阵中的连接)
    contacts <- which(ckm_network[i, ] == 1)
    if (length(contacts) > 0) {
      # 本月之前采用的联系人数量
      contacts_adopted_before <- sum(ckm_nodes$adoption_date[contacts] < month,
                                     na.rm = TRUE)
      # 本月及之前采用的联系人数量
      contacts_adopted_up_to_now <- sum(ckm_nodes$adoption_date[contacts] <=
                                          month, na.rm = TRUE)
      doctor_month_data$contacts_before[row_idx] <- contacts_adopted_before
      doctor_month_data$contacts_up_to_now[row_idx] <-
        contacts_adopted_up_to_now
    } else {
      doctor_month_data$contacts_before[row_idx] <- 0
      doctor_month_data$contacts_up_to_now[row_idx] <- 0
    }
  }
}
# 解释为什么应该有 6 列和 2125 行
cat("数据框维度:", dim(doctor_month_data), "\n")
```

```
## 数据框维度: 2125 6
```

```
cat("解释：\n")
```

```
## 解释：
```

```
cat("- 6 列：doctor_id, month, started_this_month, adopted_before,
    contacts_before, contacts_up_to_now\n")
```

```
## - 6列：doctor_id, month, started_this_month, adopted_before,
##     contacts_before, contacts_up_to_now
```

```
cat("- 2125 行： ", n_doctors, "个医生 ×", n_months, "个月 =",
    n_doctors * n_months, "行\n")
```

```
## - 2125行：  125 个医生 × 17 个月 = 2125 行
```

```
head(doctor_month_data)
```

```
##   doctor_id month started_this_month adopted_before contacts_before
## 1         1     1               TRUE          FALSE               0
## 2         1     2              FALSE           TRUE               1
## 3         1     3              FALSE           TRUE               1
## 4         1     4              FALSE           TRUE               2
## 5         1     5              FALSE           TRUE               3
## 6         1     6              FALSE           TRUE               3
##   contacts_up_to_now
## 1                  1
## 2                    1
## 3                    2
## 4                    3
## 5                    3
## 6                    3
```

3. Let

$$p_k = \Pr(\text{A doctor starts prescribing tetracycline this month} \mid \text{Number of doctor's contacts prescribing before this month} = k) \tag{1}$$

and

$$q_k = \Pr(\text{A doctor starts prescribing tetracycline this month} \mid \text{Number of doctor's contacts prescribing this month} = k) \tag{2}$$

We suppose that p_k and q_k are the same for all months.

a. Explain why there should be no more than 21 values of k for which we can estimate p_k and q_k directly from the data.

```r
# 检查网络中每个医生的最大联系人数
max_contacts <- max(rowSums(ckm_network))
max_k_before <- max(doctor_month_data$contacts_before, na.rm = TRUE)
max_k_up_to_now <- max(doctor_month_data$contacts_up_to_now, na.rm = TRUE)

cat("网络分析结果：\n")
```

```
## 网络分析结果：
```

```r
cat("- 网络中医生的最大联系人数:", max_contacts, "\n")
```

```
## - 网络中医生的最大联系人数: 20
```

```
cat("- 实际数据中 contacts_before 的最大值:", max_k_before, "\n")
```

```
## - 实际数据中contacts_before的最大值： 18
```

```
cat("- 实际数据中 contacts_up_to_now 的最大值:", max_k_up_to_now, "\n\n")
```

```
## - 实际数据中contacts_up_to_now的最大值： 18
```

```
cat("\n解释：\n")
```

```
##
## 解释：
```

```
cat("- 应该不超过 21 个 k 值的原因是网络中任何医生的联系人数量都不会超过", max_contacts, "\n"
```

```
## - 应该不超过21个k值的原因是网络中任何医生的联系人数量都不会超过 20
```

```
cat("- 因此 k（已采用联系人的数量）不可能超过医生的总联系人数\n")
```

```
## - 因此k（已采用联系人的数量）不可能超过医生的总联系人数
```

```
cat("- 实际数据显示最大 k 值约为", max(max_k_before, max_k_up_to_now), "\n")
```

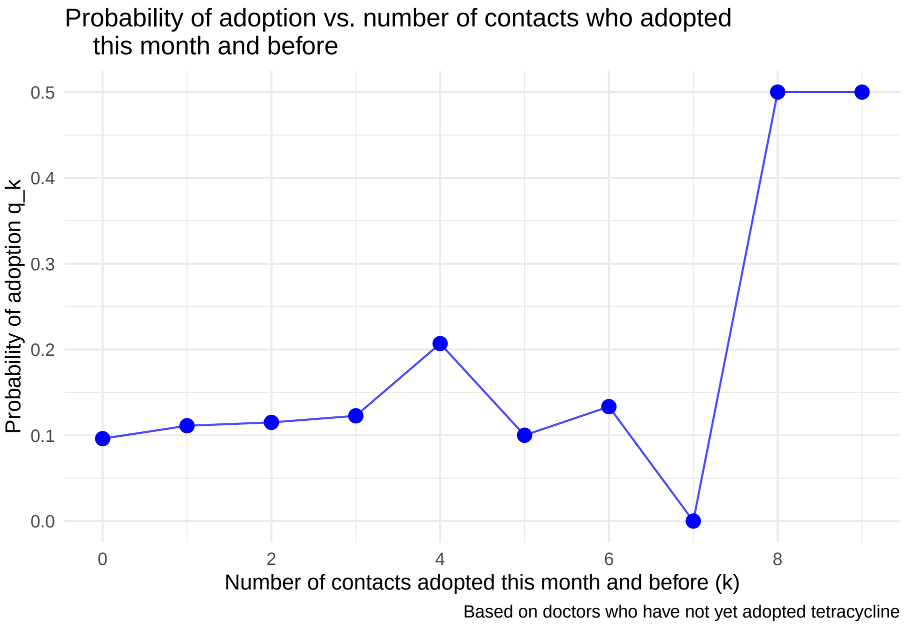
```
## - 实际数据显示最大k值约为 18
```

b. Create a vector of estimated p_k probabilities, using the data frame from (2). Plot the probabilities against the number of prior-adoptee contacts k.

```
# 计算 p_k: 基于本月之前采用的联系人数量
# 只考虑尚未采用的医生（adopted_before = FALSE）
eligible_data <- doctor_month_data[!doctor_month_data$adopted_before, ]
# 按 contacts_before 分组计算概率
p_k_summary <- eligible_data %>%
  group_by(contacts_before) %>%
  summarise(
    n_observations = n(),                                    # 观察数量
    n_started = sum(started_this_month, na.rm = TRUE),       # 开始采用的数量
    p_k = n_started / n_observations,                        # 概率估计
    .groups = 'drop'
  ) %>%
  arrange(contacts_before)

# 创建完整的 p_k 向量 (包括没有观察到的 k 值)
max_k <- max(p_k_summary$contacts_before)
p_k_vector <- rep(NA, max_k + 1)
names(p_k_vector) <- 0:max_k

for (i in 1:nrow(p_k_summary)) {
  k_val <- p_k_summary$contacts_before[i]
  p_k_vector[k_val + 1] <- p_k_summary$p_k[i]  # +1 因为索引从 1 开始
}

print(p_k_summary)
```

```
## # A tibble: 10 x 4
##    contacts_before n_observations n_started    p_k
##              <dbl>          <int>     <int>  <dbl>
##  1               0            406        39 0.0961
##  2               1            198        22 0.111
##  3               2            200        23 0.115
##  4               3            106        13 0.123
##  5               4             29         6 0.207
##  6               5             20         2 0.1
##  7               6             15         2 0.133
##  8               7              3         0 0
##  9               8              2         1 0.5
## 10               9              2         1 0.5
```

```
# 绘制 p_k 概率图
library(ggplot2)
ggplot(p_k_summary, aes(x = contacts_before, y = p_k)) +
  geom_point(size = 3, color = "blue") +
  geom_line(color = "blue", alpha = 0.7) +
  labs(
    title = "Probability of adoption vs. number of contacts who adopted
    this month and before", # 采用概率 vs 本月及之前采用的联系人数量
    x = "Number of contacts adopted this month and before (k)", # 本月及之前采用的联系人数
    y = "Probability of adoption q_k", # 采用概率
    caption = "Based on doctors who have not yet adopted tetracycline"
    # 基于尚未采用四环素的医生
  ) +
  theme_minimal() +
  scale_x_continuous(breaks = seq(0, max_k, 2))
```

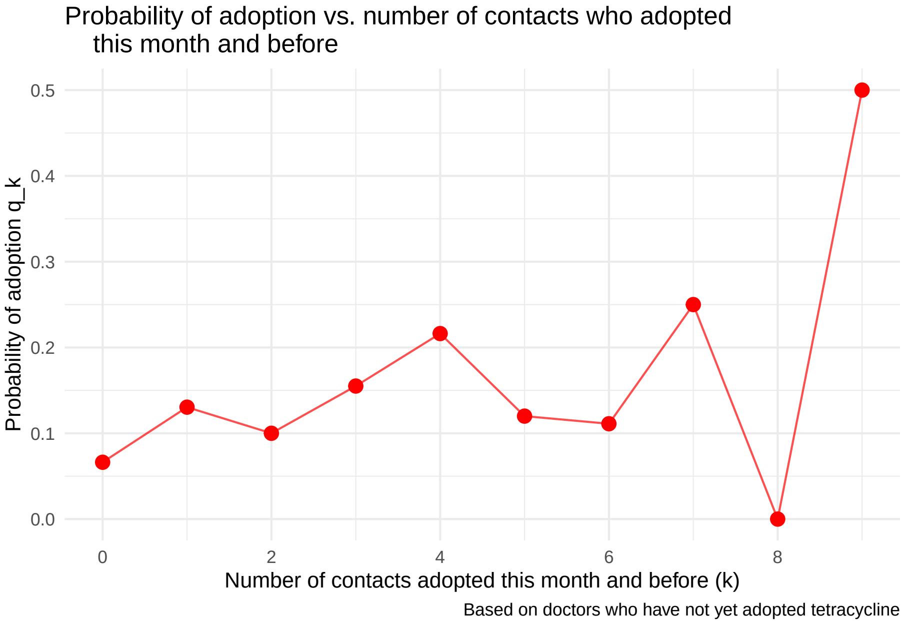


c. Create a vector of estimated q_k probabilities, using the data frame from (2). Plot the probabilities against the number of prior-or-contemporary-adoptee contacts k.

```
# 计算 q_k: 基于本月及之前采用的联系人数量
# 同样只考虑尚未采用的医生
q_k_summary <- eligible_data %>%
  group_by(contacts_up_to_now) %>%
  summarise(
    n_observations = n(),                                # 观察数量
    n_started = sum(started_this_month, na.rm = TRUE),      # 开始采用的数量
    q_k = n_started / n_observations,                       # 概率估计
    .groups = 'drop'
  ) %>%
  arrange(contacts_up_to_now)

# 创建完整的 q_k 向量
max_k_q <- max(q_k_summary$contacts_up_to_now)
q_k_vector <- rep(NA, max_k_q + 1)
names(q_k_vector) <- 0:max_k_q

for (i in 1:nrow(q_k_summary)) {
  k_val <- q_k_summary$contacts_up_to_now[i]
  q_k_vector[k_val + 1] <- q_k_summary$q_k[i]
}

print(q_k_summary)
```

```
## # A tibble: 10 x 4
##    contacts_up_to_now n_observations n_started    q_k
##                 <dbl>          <int>     <int>  <dbl>
##  1                  0            302        20 0.0662
##  2                  1            230        30 0.130
##  3                  2            230        23 0.1
##  4                  3            129        20 0.155
##  5                  4             37         8 0.216
##  6                  5             25         3 0.12
##  7                  6             18         2 0.111
##  8                  7              4         1 0.25
##  9                  8              2         0 0
## 10                  9              4         2 0.5
```

```
# 绘制 q_k 概率图
ggplot(q_k_summary, aes(x = contacts_up_to_now, y = q_k)) +
  geom_point(size = 3, color = "red") +
  geom_line(color = "red", alpha = 0.7) +
  labs(
    title = "Probability of adoption vs. number of contacts who adopted
    this month and before", # 采用概率 vs 本月及之前采用的联系人数量
    x = "Number of contacts adopted this month and before (k)", # 本月及之前采用的联系人数
    y = "Probability of adoption q_k", # 采用概率
    caption = "Based on doctors who have not yet adopted tetracycline"
    # 基于尚未采用四环素的医生
  ) +
  theme_minimal() +
  scale_x_continuous(breaks = seq(0, max_k_q, 2))
```



4. Because it only conditions on information from the previous month, p_k is a little easier to interpret than q_k. It is the probability per month

that a doctor adopts tetracycline, if they have exactly k contacts who had already adopted tetracycline.

a. Suppose $p_k = a + bk$. This would mean that each friend who adopts the new drug increases the probability of adoption by an equal amount. Estimate this model by least squares, using the values you constructed in (3b). Report the parameter estimates.

```
# 线性模型: p_k = a + bk
# 确保只使用有观察数据的 k 值，且观察数足够多的点
model_data <- p_k_summary %>%
  filter(!is.na(p_k) & n_observations >= 5)  # 至少 5 个观察

# 使用加权最小二乘法 (WLS)，将观测数作为权重
linear_model_weighted <- lm(p_k ~ contacts_before,
                            data = model_data,
                            weights = n_observations)
# 报告参数估计
summary(linear_model_weighted)
```

```
##
## Call:
## lm(formula = p_k ~ contacts_before, data = model_data, weights = n_observations)
##
## Weighted Residuals:
##        1        2        3        4        5        6        7
## -0.03142  0.05260 -0.03008 -0.04365  0.37837 -0.20746 -0.08834
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     0.097619   0.008604  11.346  9.3e-05 ***
## contacts_before 0.009754   0.004567   2.136   0.0858 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 0.2003 on 5 degrees of freedom
## Multiple R-squared:  0.477,  Adjusted R-squared:  0.3724
## F-statistic: 4.561 on 1 and 5 DF,  p-value: 0.0858
```

```
# 提取参数
a_linear <- coef(linear_model_weighted)[1]  # 截距
b_linear <- coef(linear_model_weighted)[2]  # 斜率

cat("加权线性模型参数估计：\n")
```

```
## 加权线性模型参数估计：
```

```
cat("a（截距）=", round(a_linear, 6), "\n")
```

```
## a（截距）= 0.097619
```

```
cat("b（斜率）=", round(b_linear, 6), "\n")
```

```
## b（斜率）= 0.009754
```

```
cat("解释：每增加一个已采用的联系人，采用概率增加", round(b_linear, 6), "\n")
```

```
## 解释：每增加一个已采用的联系人，采用概率增加 0.009754
```

b. Suppose $p_k = e^{a+bk}/(1+e^{a+bk})$. Explain, in words, what this model would imply about the impact of adding one more adoptee friend on a given doctor's probability of adoption. (You can suppose that $b > 0$, if that makes it easier.) Estimate the model by least squares, using the values you constructed in (3b).

```
# Logistic 模型的理论解释
cat("Logistic模型的含义：\n")
```

```
## Logistic模型的含义：
```

```
cat("当 p_k = e^(a+bk)/(1+e^(a+bk)) 且 b > 0 时：\n")
```

```
## 当 p_k = e^(a+bk)/(1+e^(a+bk)) 且 b > 0 时：
```

```
cat("1. 概率始终在 0 和 1 之间，符合概率的定义\n")
```

```
## 1. 概率始终在0和1之间，符合概率的定义
```

```
cat("2. 每增加一个采用的朋友都会增加采用概率\n")
```

```
## 2. 每增加一个采用的朋友都会增加采用概率
```

```
cat("3. 边际效应递减：\n")
```

```
## 3. 边际效应递减：
```

```
cat("   - 当基础概率很低时，新朋友的影响较小\n")
```

```
##    - 当基础概率很低时，新朋友的影响较小
```

```
cat("   - 当基础概率接近 0.5 时，新朋友的影响最大\n")
```

```
##    - 当基础概率接近0.5时，新朋友的影响最大
```

```
cat("   - 当基础概率很高时，新朋友的影响又较小\n")
```

```
##    - 当基础概率很高时，新朋友的影响又较小
```

```
cat("4. 这符合社会影响的现实：极端情况下影响力有限\n\n")
```

```
## 4. 这符合社会影响的现实：极端情况下影响力有限
```

```
# 拟合 logistic 模型
model_data_logit <- model_data
# 首先准备数据：需要采用者数量和未采用者数量
model_data_glm <- p_k_summary %>%
  filter(!is.na(p_k) & n_observations >= 5) %>%
  mutate(
    adopters = round(p_k * n_observations),
    non_adopters = n_observations - adopters
  )

# 使用 GLM 进行 logistic 回归
```

```
logistic_model_glm <- glm(cbind(adopters, non_adopters) ~ contacts_before,
                          family = binomial,
                          data = model_data_glm)

summary(logistic_model_glm)
```

```
##
## Call:
## glm(formula = cbind(adopters, non_adopters) ~ contacts_before,
##     family = binomial, data = model_data_glm)
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -2.21749    0.14120 -15.705   <2e-16 ***
## contacts_before  0.09426    0.06907   1.365    0.172
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 3.4364  on 6  degrees of freedom
## Residual deviance: 1.6364  on 5  degrees of freedom
## AIC: 33.412
##
## Number of Fisher Scoring iterations: 4
```

```
a_logistic_glm <- coef(logistic_model_glm)[1]
b_logistic_glm <- coef(logistic_model_glm)[2]

# 提取参数
cat("a =", round(a_logistic_glm, 6), "\n")
```

```
## a = -2.217486
```

```
cat("b =", round(b_logistic_glm, 6), "\n")
```

```
## b = 0.094263
```

```
cat("GLM Logistic 模型参数估计 (通过 logit 变换): \n")
```

```
## GLM Logistic模型参数估计（通过logit变换）：
```

```
cat("a =", round(a_logistic_glm, 6), "\n")
```

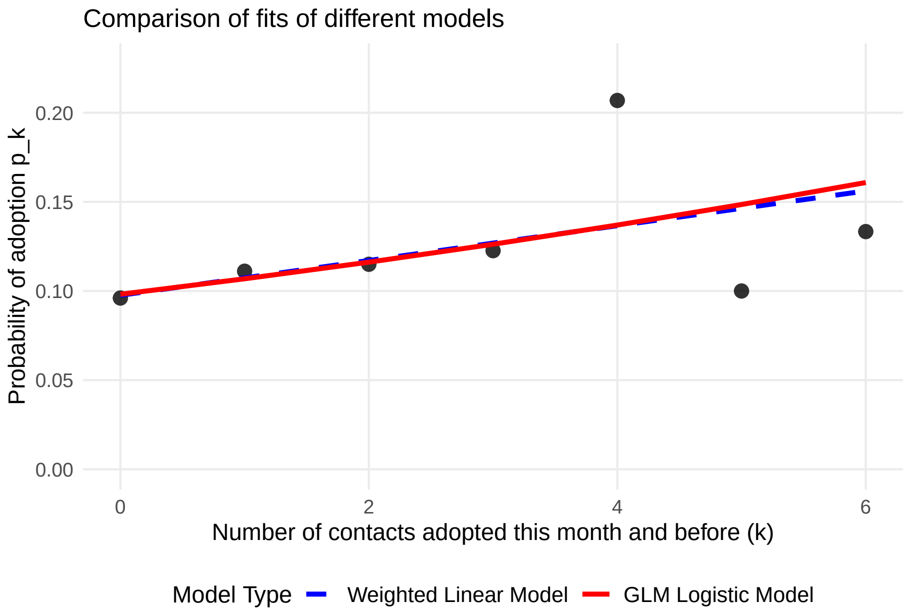
```
## a = -2.217486
```

```
cat("b =", round(b_logistic_glm, 6), "\n")
```

```
## b = 0.094263
```

c. Plot the values from (3b) along with the estimated curves from (4a) and (4b). (You should have one plot, with k on the horizontal axis, and probabilities on the vertical axis .) Which model do you prefer, and why?

```
# 创建预测值
k_range <- 0:max(model_data$contacts_before)

# 加权线性模型预测
linear_pred <- a_linear + b_linear * k_range
# 确保概率不会小于 0 或大于 1
linear_pred <- pmax(0, pmin(1, linear_pred))

# GLM Logistic 模型预测
# glm 的预测需要指定 type="response" 来获得概率
logistic_pred_glm <- predict(logistic_model_glm,
                             newdata = data.frame(contacts_before = k_range),
                             type = "response")

# 创建绘图数据
plot_data <- data.frame(
  k = k_range,
  linear = linear_pred,
  logistic_glm = logistic_pred_glm
)

# 绘制比较图
library(ggplot2)
library(tidyr)

# 重塑数据以便于绘图
plot_data_long <- plot_data %>%
  pivot_longer(cols = c(linear, logistic_glm),
               names_to = "model",
               values_to = "predicted")

p <- ggplot() +
  # 原始数据点
  geom_point(data = model_data, aes(x = contacts_before, y = p_k),
             size = 3, color = "black", alpha = 0.8) +
  # 模型预测线
  geom_line(data = plot_data_long, aes(x = k, y = predicted,
                                       color = model, linetype = model),
            linewidth = 1.2) +
  scale_color_manual(name = "Model Type", # 模型类型
    values = c("linear" = "blue", "logistic_glm" = "red"),
    labels = c("Weighted Linear Model", "GLM Logistic Model") # 加权线性，GLM 逻辑回归
  ) +
  scale_linetype_manual(name = "Model Type", # 模型类型
                        values = c("linear" = "dashed", "logistic_glm" = "solid"),
                        labels = c("Weighted Linear Model",
                                   "GLM Logistic Model")) +
  # 标题和轴标签
  labs(
    title = "Comparison of fits of different models", # 不同模型的拟合比较
    x = "Number of contacts adopted this month and before (k)", # 本月之前采用的联系人数量
    y = "Probability of adoption p_k", # 采用概率
  ) +
  theme_minimal() +
  theme(legend.position = "bottom",
    legend.title = element_text(size = 12),
    legend.text = element_text(size = 11),
    axis.title = element_text(size = 12),
    axis.text = element_text(size = 10),
    panel.grid.minor = element_blank()
  ) +
  # 设置 y 轴范围
  ylim(0, max(c(model_data$p_k, linear_pred, logistic_pred_glm),
              na.rm = TRUE) * 1.1) # 设置 y 轴范围为 0 到最大值的 1.1 倍

# 显示图表
print(p)
```



```
# 计算拟合优度
linear_fitted <- predict(linear_model_weighted)
logistic_fitted_glm <- predict(logistic_model_glm, type = "response")

linear_rmse <- sqrt(mean((model_data$p_k - linear_fitted)^2, na.rm = TRUE))
logistic_rmse_glm <- sqrt(mean((model_data_glm$p_k - logistic_fitted_glm)^2,
                               na.rm = TRUE))

# 模型比较分析
cat("\n模型比较分析：\n")
```

```
##
## 模型比较分析：
```

```
cat("加权线性模型RMSE:", round(linear_rmse, 6), "\n")
```

```
## 加权线性模型RMSE: 0.033054
```

```
cat("GLM Logistic模型RMSE:", round(logistic_rmse_glm, 6), "\n")
```

```
## GLM Logistic模型RMSE: 0.03388
```

```
cat("\n模型选择建议：\n")
```

```
##
## 模型选择建议：
```

```
if (logistic_rmse_glm < linear_rmse) {
  cat("- Logistic模型具有更小的RMSE，拟合效果更好\n")
} else {
  cat("- 线性模型具有更小的RMSE，拟合效果更好\n")
}
```

```
## - 线性模型具有更小的RMSE，拟合效果更好
```

```
cat("\n各模型的优缺点：\n")
```

```
##
```

```
## 各模型的优缺点：
```

```r
cat("线性模型优势：简单易解释，计算方便\n")
```

```
## 线性模型优势：简单易解释，计算方便
```

```r
cat("线性模型劣势：可能产生负概率或大于 1 的概率\n")
```

```
## 线性模型劣势：可能产生负概率或大于1的概率
```

```r
cat("Logistic 模型优势：概率始终在 [0,1] 范围内，边际效应递减符合社会影响理论，S 型曲线符合扩散
```

```
## Logistic模型优势：概率始终在[0,1]范围内，边际效应递减符合社会影响理论，S型曲线符合扩
```

```r
cat("Logistic 模型劣势：相对复杂，解释稍难\n\n")
```

```
## Logistic模型劣势：相对复杂，解释稍难
```

```r
cat("推荐：考虑到概率的本质和社会扩散理论，建议使用 Logistic 模型。\n")
```

```
## 推荐：考虑到概率的本质和社会扩散理论，建议使用Logistic模型。
```

For quibblers, pedants, and idle hands itching for work to do: The p_k values from problem 3 aren't all equally precise, because they come from different numbers of observations. Also, if each doctor with k adoptee contacts is independently deciding whether or not to adopt with probability p_k, then the variance in the number of adoptees will depend on p_k. Say that the actual proportion who decide to adopt is $\hat{p}_k$. A little probability (exercise!) shows that in this situation, $\mathbb{E}[\hat{p}_k] = p_k$, but that $\mathrm{Var}[\hat{p}_k] = p_k(1-p_k)/n_k$, where n_k is the number of doctors in that situation. (We estimate probabilities more precisely when they're really extreme [close to 0 or 1], and/or we have lots of observations.) We can estimate that variance as $\hat{V}_k = \hat{p}_k(1-\hat{p}_k)/n_k$. Find the $\hat{V}_k$, and then re-do the estimation in (4a) and (4b) where the squared error for p_k is divided by $\hat{V}_k$. How much do the parameter estimates change? How much do the plotted curves in (4c) change?

```r
# 额外问题：加权最小二乘法
cat("额外分析：考虑观察精度的加权最小二乘法\n\n")
```

```
## 额外分析：考虑观察精度的加权最小二乘法
```

```
# 计算每个 p_k 的方差估计
model_data_weighted <- model_data %>%
  mutate(
    # 计算方差：V_k = p_k(1-p_k)/n_k
    variance = p_k * (1 - p_k) / n_observations,
    # 权重是方差的倒数
    weight = 1 / variance
  )

# 处理方差为 0 的情况（当 p_k = 0 或 1 时）
model_data_weighted$weight[is.infinite(model_data_weighted$weight) |
                             is.na(model_data_weighted$weight)] <- 1

print(model_data_weighted)
```

```
## # A tibble: 7 x 6
##   contacts_before n_observations n_started    p_k variance weight
##             <dbl>          <int>     <int>  <dbl>    <dbl>  <dbl>
## 1               0            406        39 0.0961 0.000214  4676.
## 2               1            198        22 0.111  0.000499  2005.
## 3               2            200        23 0.115  0.000509  1965.
## 4               3            106        13 0.123  0.00102    985.
## 5               4             29         6 0.207  0.00566    177.
## 6               5             20         2 0.1    0.0045     222.
## 7               6             15         2 0.133  0.00770    130.
```

```
# 新的加权线性回归（使用方差权重）
variance_weighted_linear_model <- lm(p_k ~ contacts_before,
                                     data = model_data_weighted,
                                     weights = weight)

# 新的加权 logistic 回归
model_data_weighted_logit <- model_data_weighted %>%
  mutate(
    p_k_adj = ifelse(p_k == 0, 0.001, ifelse(p_k == 1, 0.999, p_k)),
    logit_p = log(p_k_adj / (1 - p_k_adj))
  )
variance_weighted_logit_model <- lm(logit_p ~ contacts_before,
                                    data = model_data_weighted_logit,
                                    weights = weight)

# 比较参数估计
cat("参数估计比较：\n")
```

```
## 参数估计比较：
```

```
cat("线性模型：\n")
```

```
## 线性模型：
```

```
cat("  观测数权重：a =", round(a_linear, 6), ", b =", round(b_linear, 6), "\n")
```

```
##   观测数权重：a = 0.097619 , b = 0.009754
```

```
cat("  方差权重：a =", round(coef(variance_weighted_linear_model)[1], 6),
    ", b =", round(coef(variance_weighted_linear_model)[2], 6), "\n")
```

```
##   方差权重：a = 0.098107 , b = 0.008539
```

```
cat("Logistic 模型：\n")
```

```
## Logistic模型：
```

```
cat("  GLM 方法：a =", round(a_logistic_glm, 6), ", b =",
    round(b_logistic_glm, 6), "\n")
```

```
##   GLM方法：a = -2.217486 , b = 0.094263
```

```
cat("  方差权重 logit：a =", round(coef(variance_weighted_logit_model)[1], 6),
    ", b =", round(coef(variance_weighted_logit_model)[2], 6), "\n")
```

```
##   方差权重logit：a = -2.215182 , b = 0.081574
```

```
# 计算参数变化
linear_a_change <- abs(coef(variance_weighted_linear_model)[1] - a_linear)
linear_b_change <- abs(coef(variance_weighted_linear_model)[2] - b_linear)
logistic_a_change <- abs(coef(variance_weighted_logit_model)[1]
                         - a_logistic_glm)
logistic_b_change <- abs(coef(variance_weighted_logit_model)[2]
                         - b_logistic_glm)

cat("\n参数变化程度：\n")
```

```
##
## 参数变化程度：
```

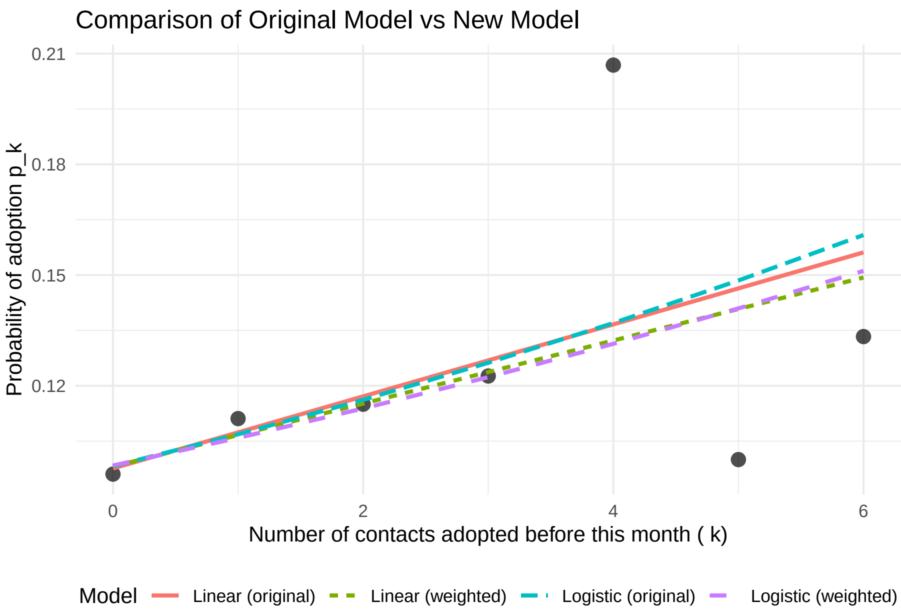
```
cat(" 线性模型参数变化： |Δa| =", round(linear_a_change, 6),
    ", |Δb| =", round(linear_b_change, 6), "\n")
```

```
## 线性模型参数变化： |Δa| = 0.000488 , |Δb| = 0.001215
```

```
cat("Logistic 模型参数变化： |Δa| =", round(logistic_a_change, 6),
    ", |Δb| =", round(logistic_b_change, 6), "\n")
```

```
## Logistic模型参数变化： |Δa| = 0.002304 , |Δb| = 0.012689
```

```
# 绘制比较图
variance_weighted_linear_pred <- coef(variance_weighted_linear_model)[1] +
  coef(variance_weighted_linear_model)[2] * k_range

variance_weighted_logistic_pred <- exp(coef(variance_weighted_logit_model)[1] +
                                        coef(variance_weighted_logit_model)[2]
                                        * k_range) /
  (1 + exp(coef(variance_weighted_logit_model)[1] +
             coef(variance_weighted_logit_model)[2] * k_range))

comparison_data <- data.frame(
  k = rep(k_range, 4),
  predicted = c(linear_pred, logistic_pred_glm,
                variance_weighted_linear_pred, variance_weighted_logistic_pred),
  model = rep(c("Linear (original)", "Logistic (original)",
                "Linear (weighted)", "Logistic (weighted)"),
              each = length(k_range))
)

ggplot() +
  geom_point(data = model_data, aes(x = contacts_before, y = p_k),
             size = 3, color = "black", alpha = 0.7) +
  geom_line(data = comparison_data, aes(x = k, y = predicted,
                                        color = model, linetype = model),
            linewidth = 1) +
  labs(
    title = "Comparison of Original Model vs New Model", # 原始模型 vs 新模型比较
    x = "Number of contacts adopted before this month ( k)", # 本月之前采用的联系人数量
    y = "Probability of adoption p_k", # 采用概率
    color = "Model", # 模型
    linetype = "Model" # 模型
  ) +
  theme_minimal() +
  theme(legend.position = "bottom")
```



```
cat("\n结论：\n")
```

```
##
## 结论：
```

```
if (max(linear_a_change, linear_b_change, logistic_a_change,
        logistic_b_change) < 0.01) {
  cat("- 参数估计变化很小，说明权重对结果影响不大\n")
} else {
  cat("- 参数估计有明显变化，说明考虑观察精度很重要\n")
}
```

```
## - 参数估计有明显变化，说明考虑观察精度很重要
```

```
cat("- 加权回归考虑了不同观察的精度差异，理论上更准确\n")
```

```
## - 加权回归考虑了不同观察的精度差异，理论上更准确
```

```
cat("- 观察数量少的组合被赋予较小权重，避免过度影响结果\n")
```

```
## - 观察数量少的组合被赋予较小权重，避免过度影响结果
```