贝叶斯分析期末课程论文

张涵 PB20010469

2023-02-18

研究的问题

2020 年夏季奥林匹克运动会，是第 32 届夏季奥林匹克运动会，于 2021 年 7 月 23 日至 8 月 8 日在日本东京都举行，为期 17 天。我在网上看到一篇文章，利用其中男子短跑 100 米这一项目之前几届的数据进行分析，进而来尝试预测这一届的比赛情况，这很契合贝叶斯统计的思想，也给了我一个练习学到的各种贝叶斯方法的机会。我在这里使用的统计工具是 Rstudio 和 rstan。

往届男子 100 米短跑数据

以下是搜集到的男子 100 米短跑历年冠军成绩，用数据框的形式直接输入。

```
## # A tibble: 29 x 6
##     Year Event    Athlete         Medal Country  Time
##    <int> <chr>    <chr>           <chr> <chr>   <dbl>
##  1  1896 100m Men Tom Burke       GOLD  USA      12
##  2  1900 100m Men Frank Jarvis    GOLD  USA      11
##  3  1904 100m Men Archie Hahn     GOLD  USA      11
##  4  1906 100m Men Archie Hahn     GOLD  USA      11.2
##  5  1908 100m Men Reggie Walker   GOLD  SAF      10.8
##  6  1912 100m Men Ralph Craig     GOLD  USA      10.8
```

```
##  7  1920 100m Men Charles Paddock GOLD  USA      10.8
##  8  1924 100m Men Harold Abrahams GOLD  GBR      10.6
##  9  1928 100m Men Percy Williams  GOLD  CAN      10.8
## 10  1932 100m Men Eddie Tolan     GOLD  USA      10.3
## # ... with 19 more rows
```

用可视化折线图的形式呈现如下：

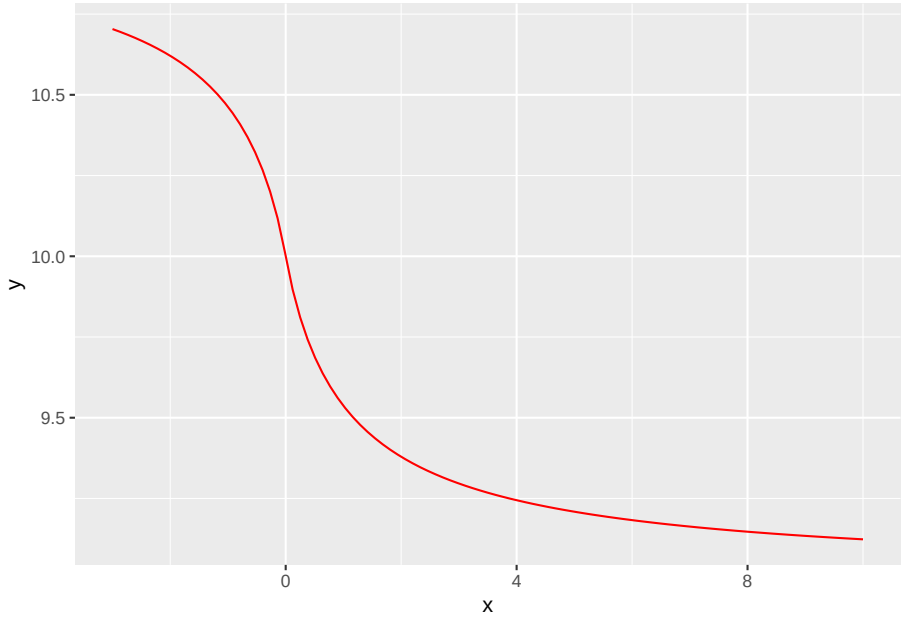
具体分步骤处理

模型取定

经过一系列数学分析，我认为男子 100 米短跑时间符合 S 型曲线形状，并且曲线的数学表达式可以大致给出

$$f(x) = L + 1 - \frac{x}{(1 + |x|^k)^{\frac{1}{k}}}$$

当 $L = 9$ 和 $k = 0.9$，其中 $f(x)$ 表示运动员成绩（time）,x 是年份（year）的某种变换，图形便是如下这个样子：



贝叶斯回归模型

从上面的数学模型出发，我们再引入一系列参数 C, S, L, k, σ 等，便得到了一个可研究的，具体的非线性贝叶斯回归模型如下：

$$
\begin{aligned}
\text{Time} &\sim \text{Normal}(\mu, \sigma) \\
\mu = f(\text{ Year}, C, S, L, k) &= L + 1 - \frac{(\text{ Year } - C)/S}{\left(1 + |(\text{ Year } - C)/S|^{k}\right)^{1/k}} \\
C &\sim \text{Normal}(1959, 5) \\
S &\sim \text{Normal}(37, 1) \\
L &\sim \text{Normal}(9, 0.2) \\
k &\sim \text{Normal}(1, 0.2) \\
\sigma &\sim \text{StudentT}(3, 0, 2.5)
\end{aligned}
$$

数据预处理

经考虑剔除掉 1896 年的记录，因为从折线图上来看这组数据过于偏离其他数据，而其他数据则一并保留。

```
## # A tibble: 28 x 6
##     Year Event    Athlete         Medal Country  Time
##    <int> <chr>    <chr>           <chr> <chr>   <dbl>
##  1  1900 100m Men Frank Jarvis    GOLD  USA      11
##  2  1904 100m Men Archie Hahn     GOLD  USA      11
##  3  1906 100m Men Archie Hahn     GOLD  USA      11.2
##  4  1908 100m Men Reggie Walker   GOLD  SAF      10.8
##  5  1912 100m Men Ralph Craig     GOLD  USA      10.8
##  6  1920 100m Men Charles Paddock GOLD  USA      10.8
##  7  1924 100m Men Harold Abrahams GOLD  GBR      10.6
##  8  1928 100m Men Percy Williams  GOLD  CAN      10.8
##  9  1932 100m Men Eddie Tolan     GOLD  USA      10.3
## 10  1936 100m Men Jesse Owens     GOLD  USA      10.3
## # ... with 18 more rows
```

Rstudio 与 rstan

这里采用了 Stan 来做 MCMC 抽样，以及进一步的模拟。是通过在 Rstudio 中调用 rstan 来实现，具体代码格式可以参看 rstan 官网，这里也并不复杂，我想谈的是一点安装 rstan 包的经验，我按照 Github 上的指南，各部分都装的最新版本，但安装后运行一直报错，主要问题出在 R 的版本，Rtools 以及 rstan 包的版本不适配，最后我选择的是 R version 4.0.2,Rtools version 4.0, 以及 rstan 包从”source” 下载最新的，可以正常运行。

```
stan_program <- "
data {
  int N;
  vector[N] year;
  vector[N] time;
```

```
}
parameters {
  real C;
  real S;
  real L;
  real k;
  real<lower=0> sigma;
}
model {
  vector[N] mu;

  for(i in 1:N) {
    mu[i] =  L + 1 - ((year[i]-C)/S) / (1+fabs((year[i]-C)/S)^k)^(1/k);
  }

  C ~ normal(1959, 5);
  S ~ normal(37, 1);
  L ~ normal(9, 0.2);
  k ~ normal(1, 0.2);
  sigma ~ student_t(3, 0, 2.5);

  time ~ normal(mu, sigma);

}
generated quantities {
  vector[N] y_rep;

  for (n in 1:N) {
    y_rep[n] = normal_rng(L + 1 - ((year[n]-C)/S) / (1+fabs((year[n]-C)/S)^k)^(1/k), si
  }

}
"
```

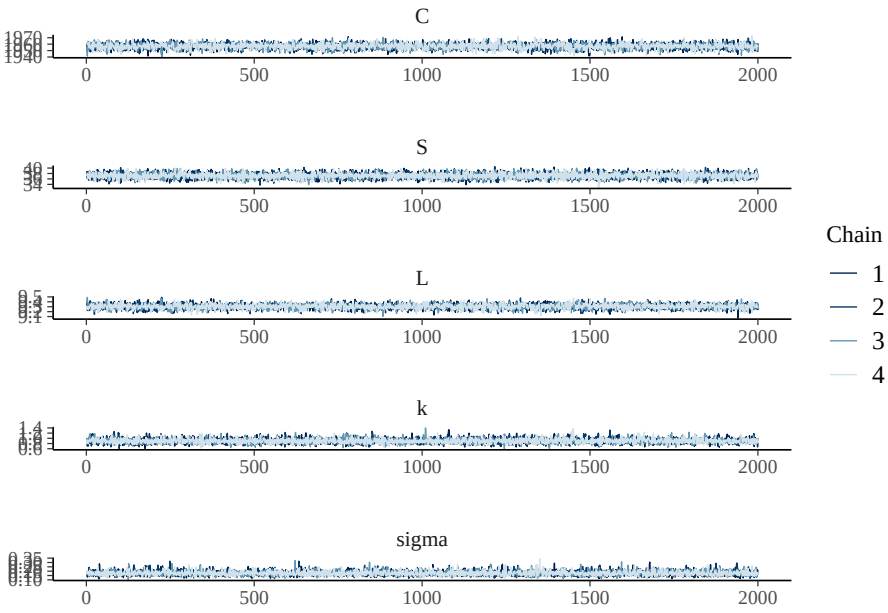
```
stan_data <- golddata1900 %>%
  tidybayes::compose_data(
    N         = nrow(.),
    year      = Year,
    time      = Time
  )



fit <- stan(model_code = stan_program, data = stan_data,
              seed = 1024,
              iter = 4000,
              warmup = 2000)
```

抽样完毕后把结果通过如下可视化的方式呈现：

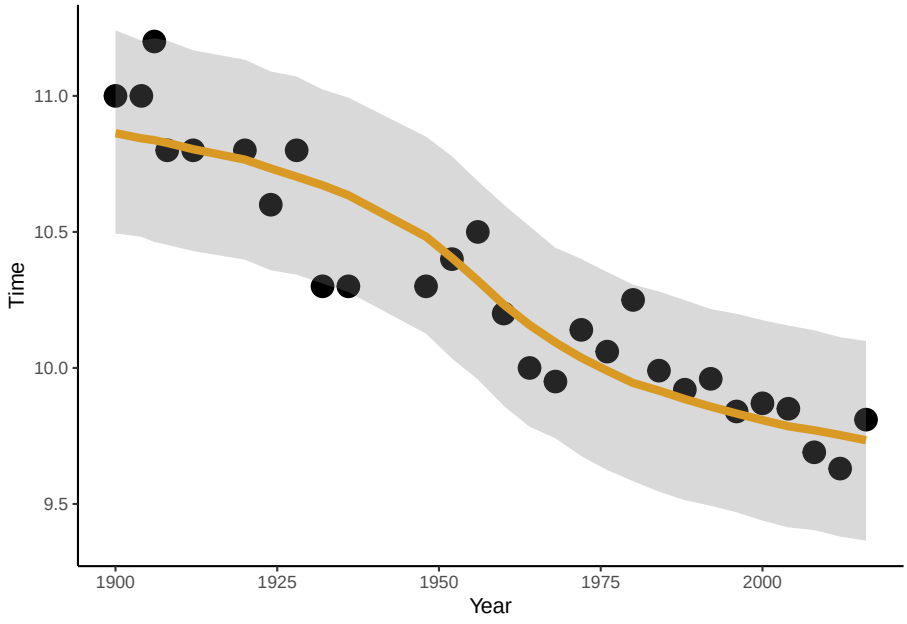
```
bayesplot::mcmc_trace(fit, pars = c("C", "S", "L", "k", "sigma"), facet_args = list(nro
```



还可伴着之前的抽样曲线在图上做出抽样点，和大致的抽样区域：

```
fit %>%
  tidybayes::gather_draws(y_rep[i]) %>%
  mean_qi() %>%
  bind_cols(golddata1900) %>%
  ggplot(aes(x = Year, y = Time)) +
  geom_point(size = 5) +
  geom_line(aes(y = .value), size = 2, color = "orange") +
  geom_ribbon(aes(ymin = .lower, ymax = .upper),
    alpha = 0.3,
    fill = "gray50"
  ) +
  theme_classic()
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
```



进一步作出预测

在训练出了想要的模型之后，便可带入时间，来对接下来几年的运动员成绩做预测了。

```
y_pred <- function(year, C, S, L, k, sigma) {
  mu <- L + 1 - ((year - C) / S) / (1 + abs((year - C) / S)^k)^(1 / k)
  rnorm(n = 1, mean = mu, sd = sigma)
}

sim <- fit %>%
  tidybayes::spread_draws(C, S, L, k, sigma) %>%
  ungroup() %>%
  rowwise() %>%
  mutate(
    pred2021 = y_pred(year = 2021, C, S, L, k, sigma),
    pred2024 = y_pred(year = 2024, C, S, L, k, sigma),
    pred2028 = y_pred(year = 2028, C, S, L, k, sigma)
  ) %>%
  ungroup()

sim %>%
  select(starts_with("pred")) %>%
  map_dfr(
   ~tidybayes::mean_hdi(.x)
  )
```

```
##          y     ymin     ymax .width .point .interval
## 1 9.721195 9.349919 10.07950   0.95   mean       hdi
## 2 9.707717 9.340749 10.07882   0.95   mean       hdi
## 3 9.695756 9.339937 10.06331   0.95   mean       hdi
```