Benchmark Plots

```
require(ggplot2)
```

```
Loading required package: ggplot2
```

```
require(ggpubr)
```

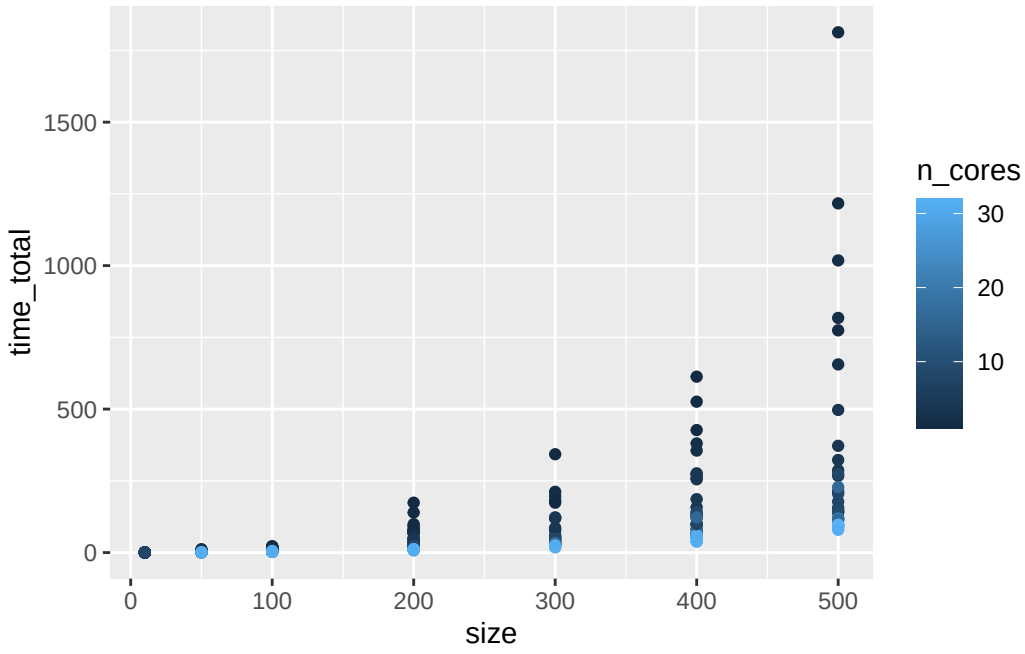
```
Loading required package: ggpubr
```

Load Timing Data

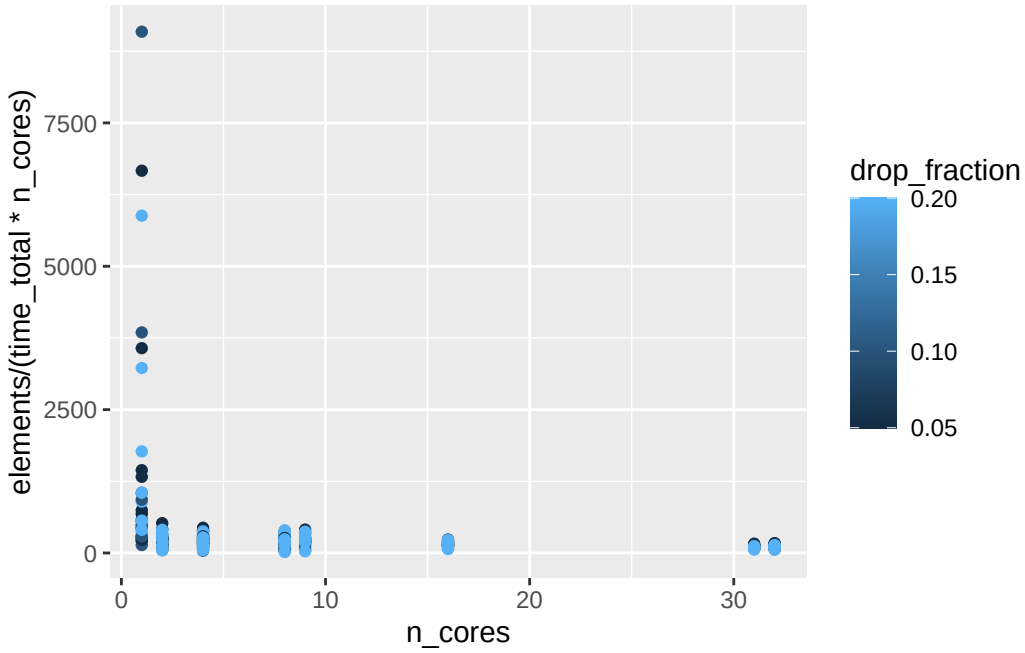
```
times_imputation <- readRDS("times_imputation.rds")
times_distance <- readRDS("times_distance.rds")
```

Plot imputation times

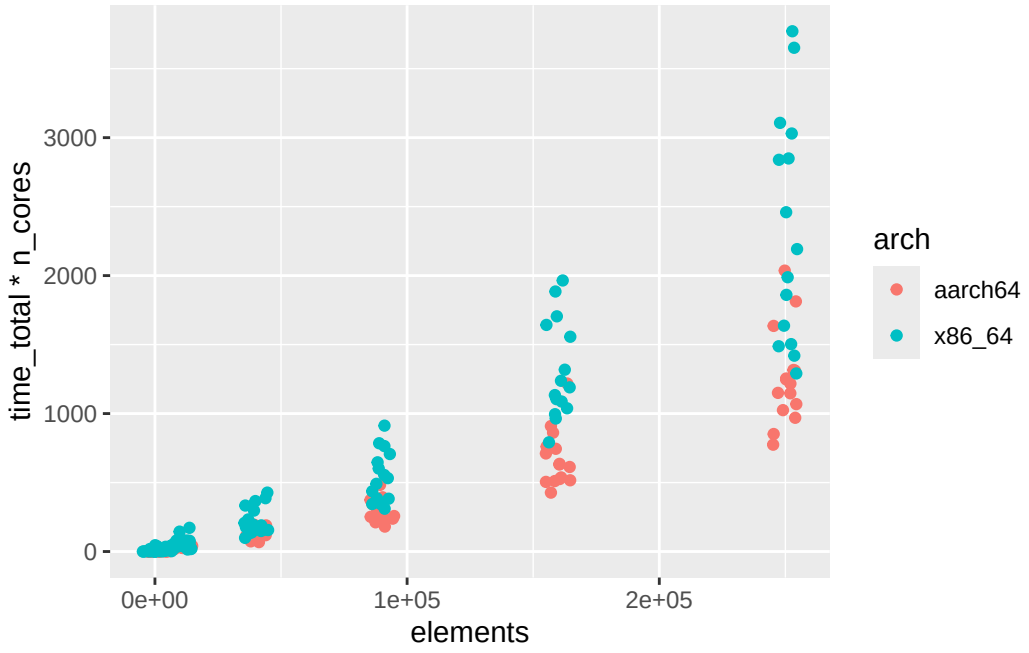
```
ggplot(times_imputation,
       aes(x=size, y=time_total, colour=n_cores)) +
  geom_point()
```



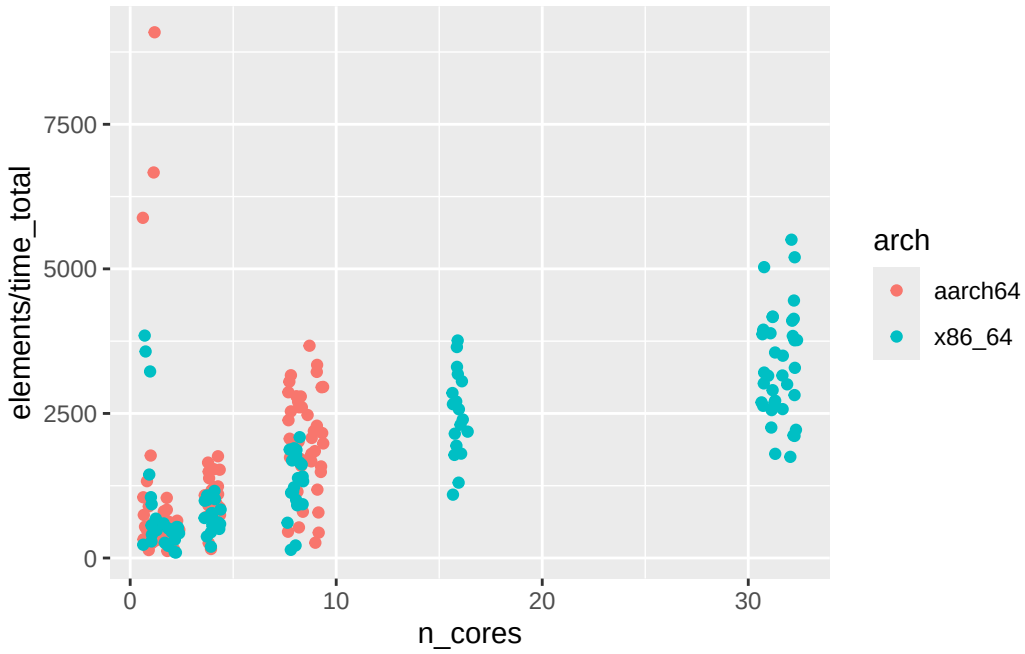

```
ggplot(times_imputation,
       aes(y=elements/(time_total*n_cores),
           x=n_cores,
           colour=drop_fraction)) +
  geom_point()
```



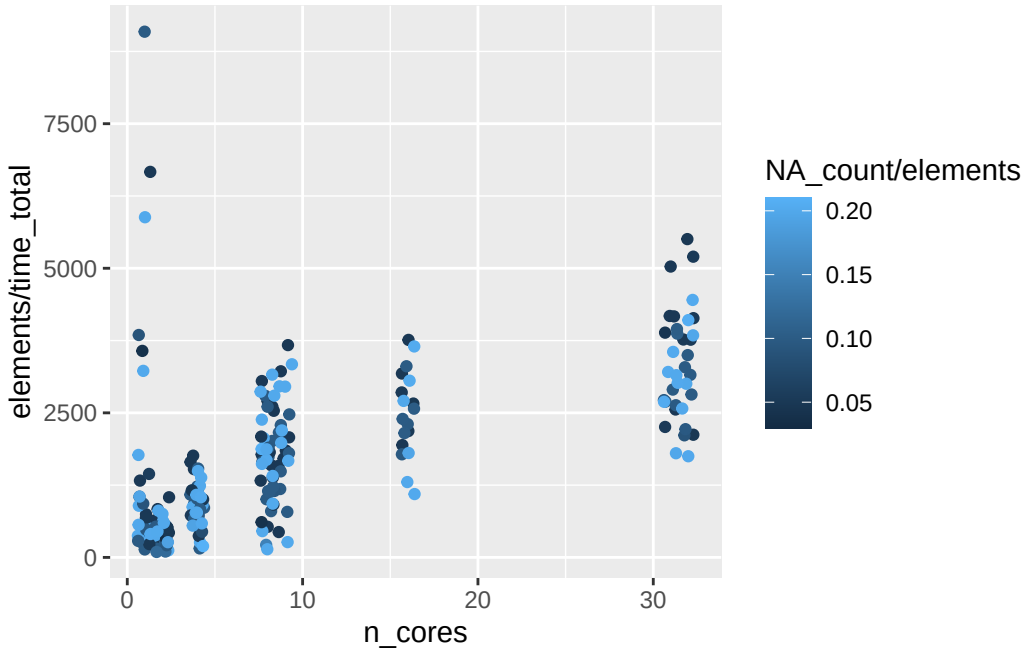
```
ggplot(times_imputation,
       aes(x=elements, y=time_total*n_cores, colour=arch)) +
  geom_jitter(width = 5000)
```



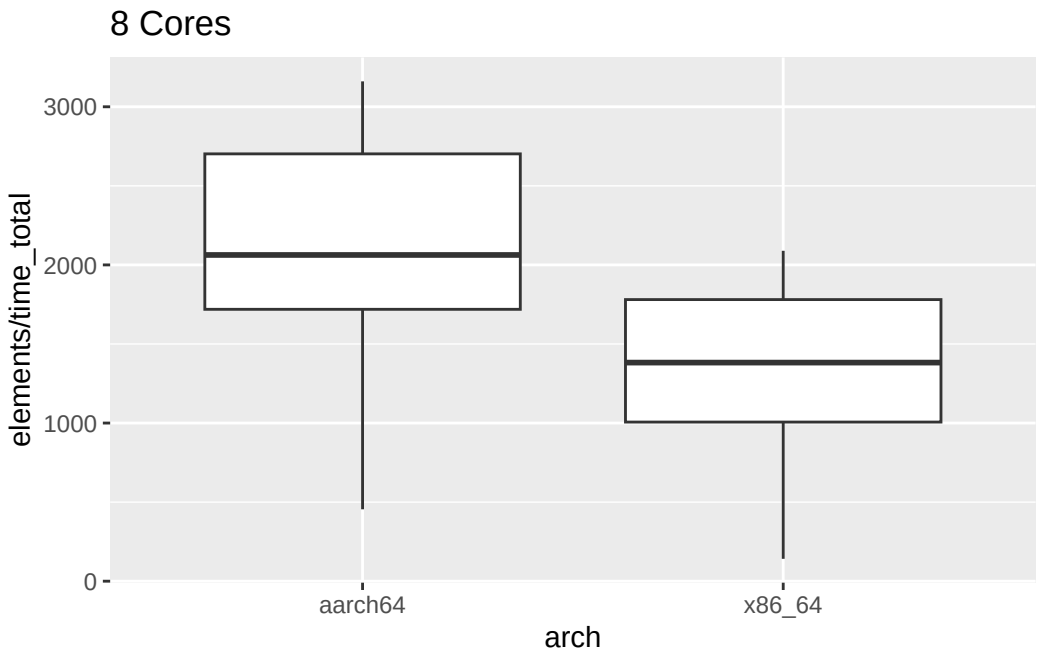

```
ggplot(times_imputation,
       aes(x=n_cores, y=elements/time_total, colour=arch)) +
  geom_point(position="jitter")
```



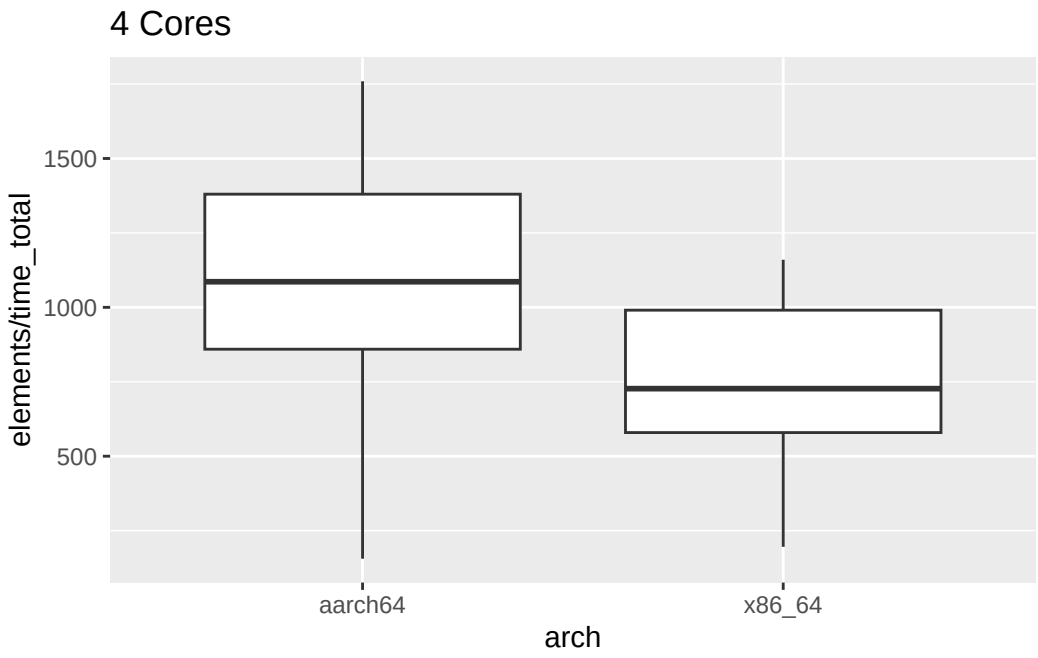

```
ggplot(times_imputation,
       aes(x=n_cores,
           y=elements/time_total,
           colour=NA_count/elements)) +
  geom_jitter()
```



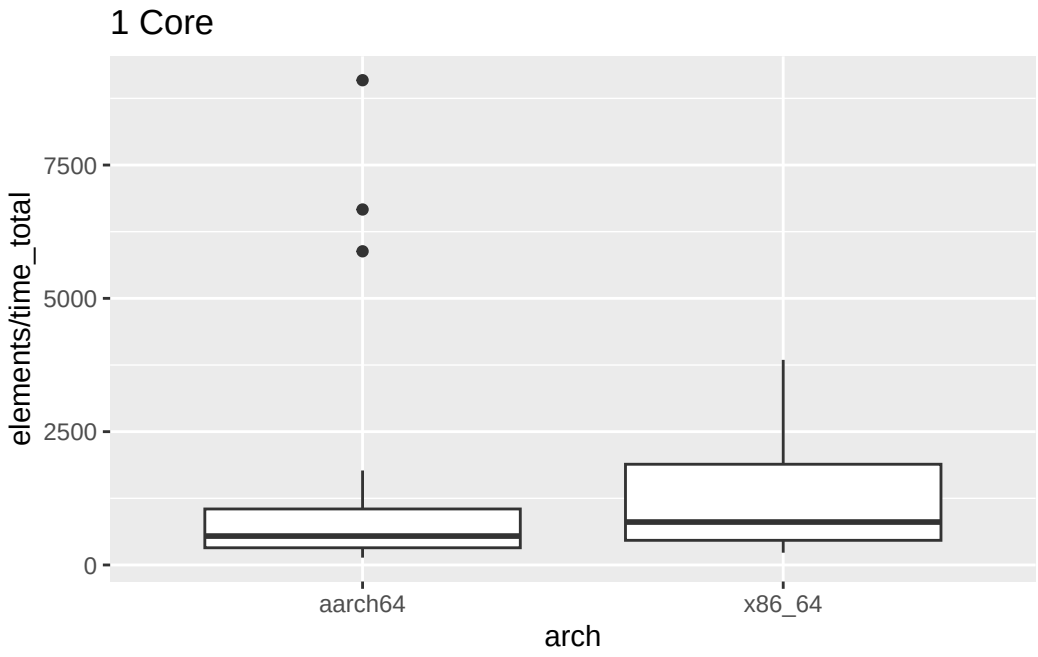
```
ggplot(times_imputation[times_imputation$n_cores==8,],
       aes(x=arch, y=elements/time_total)) +
  geom_boxplot() +
  ggtitle("8 Cores")
```



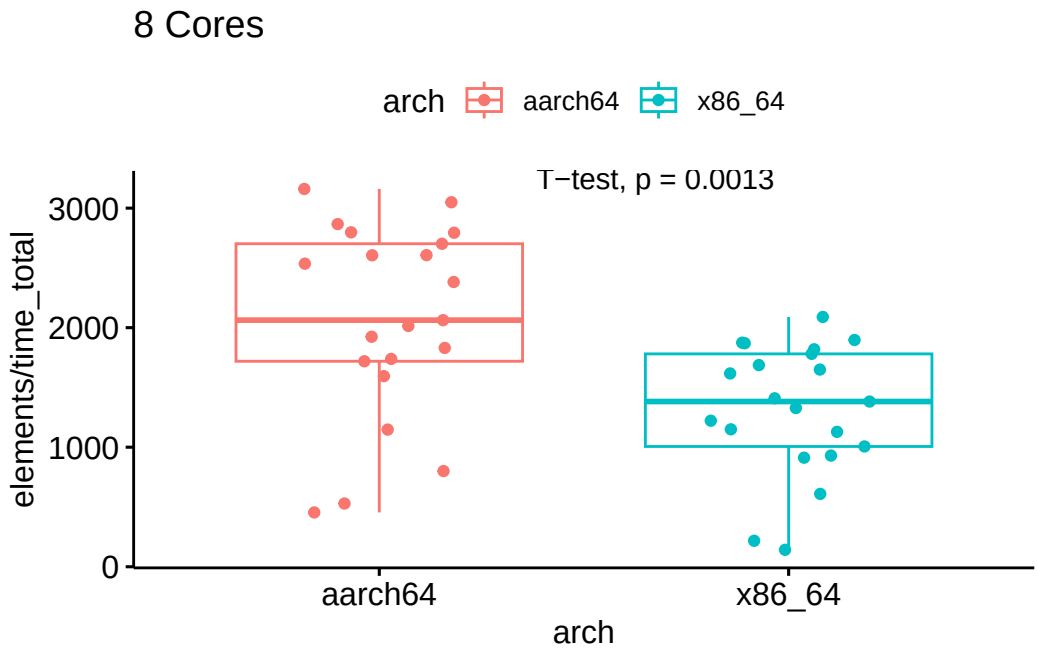

```
ggplot(times_imputation[times_imputation$n_cores==4,],
       aes(x=arch, y=elements/time_total)) +
  geom_boxplot() +
  ggtitle("4 Cores")
```



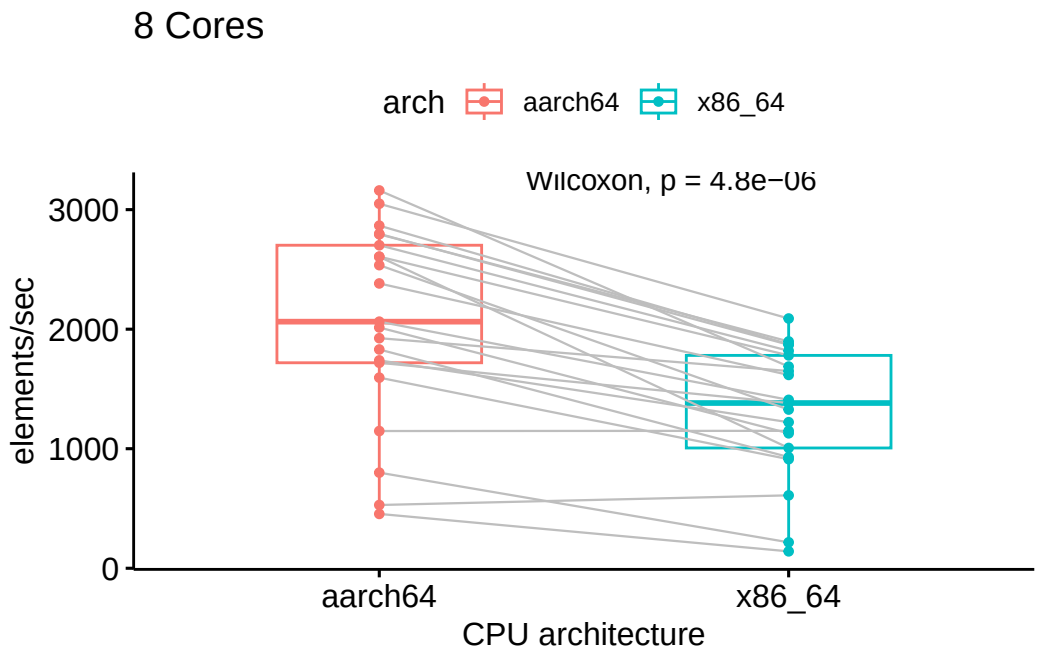

```
ggplot(times_imputation[times_imputation$n_cores==1,],
       aes(x=arch, y=elements/time_total)) +
  geom_boxplot() +
  ggtitle("1 Core")
```



```
ggboxplot(times_imputation[times_imputation$n_cores==8,],
          x="arch",
          y="elements/time_total",
          color="arch",
          add = "jitter") +
  stat_compare_means(method = "t.test", label.x = 1.5) +
  ggtitle("8 Cores")
```

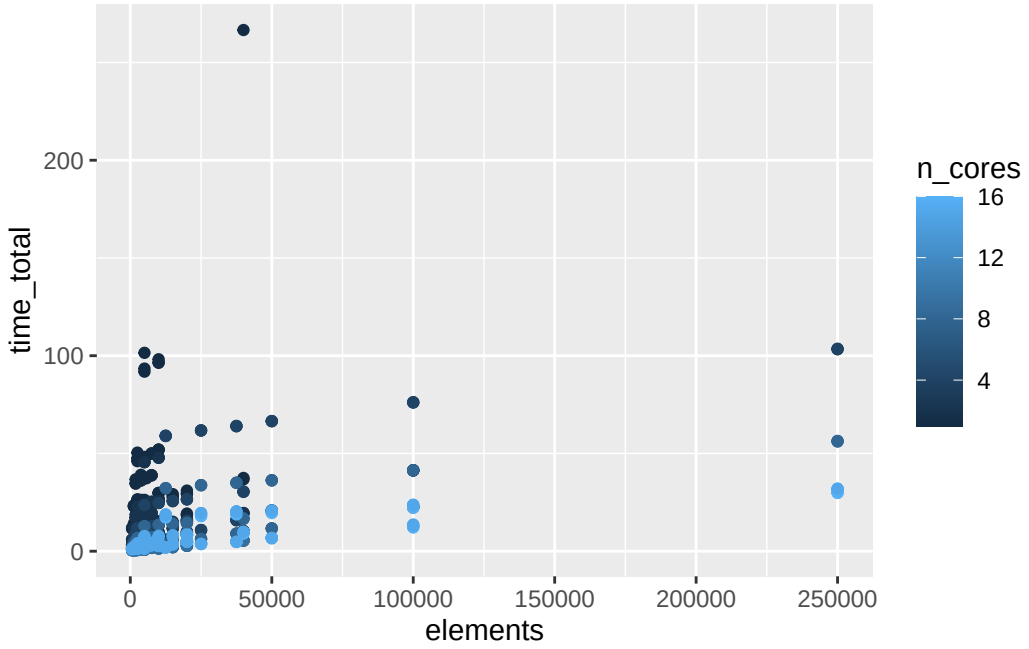



```
ggpaired(times_imputation[times_imputation$n_cores==8,],
         x="arch",
         y="elements/time_total",
         color="arch",
         line.color = "gray",
         line.size = 0.4) +
  stat_compare_means(paired = TRUE, label.x = 1.5) +
  ggtitle("8 Cores") +
  xlab("CPU architecture") +
  ylab("elements/sec")
```

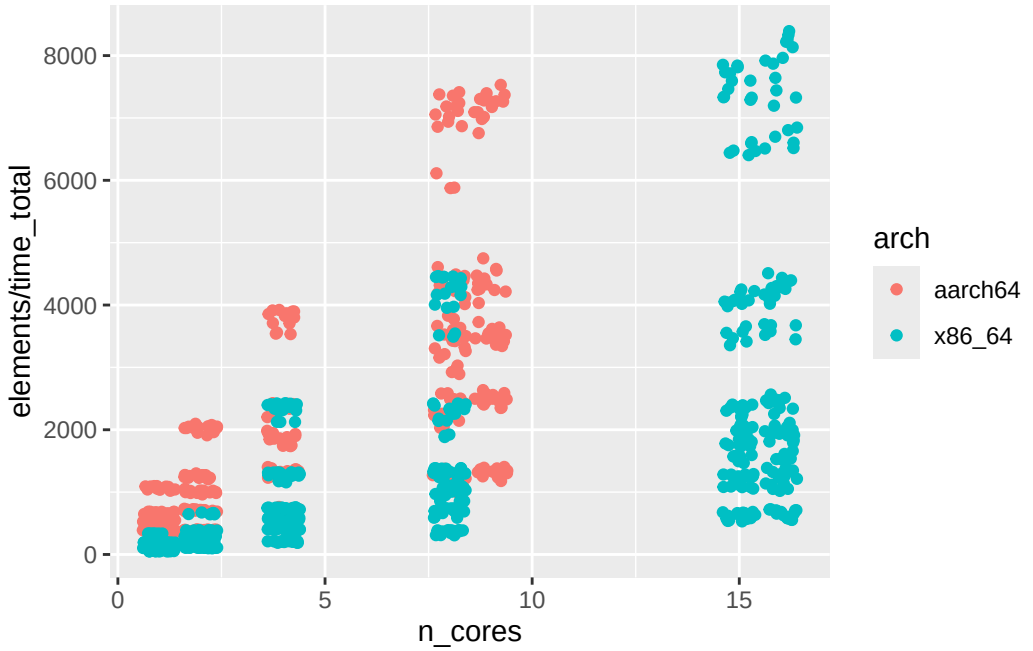


Plot Distance Times

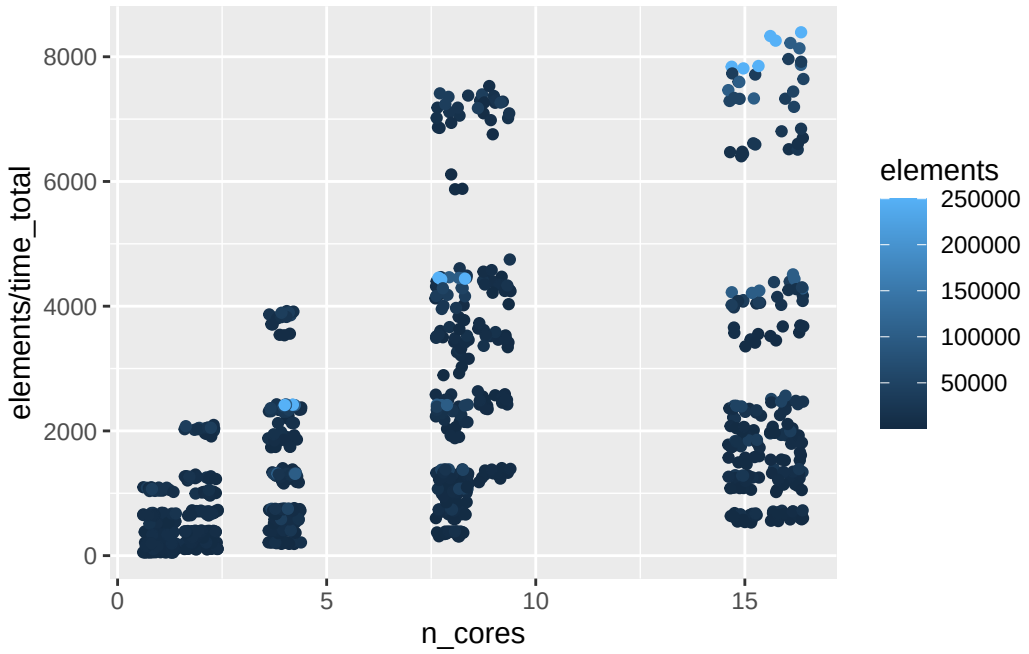
```
ggplot(times_distance, aes(x=elements, y=time_total, color=n_cores)) + geom_point()
```



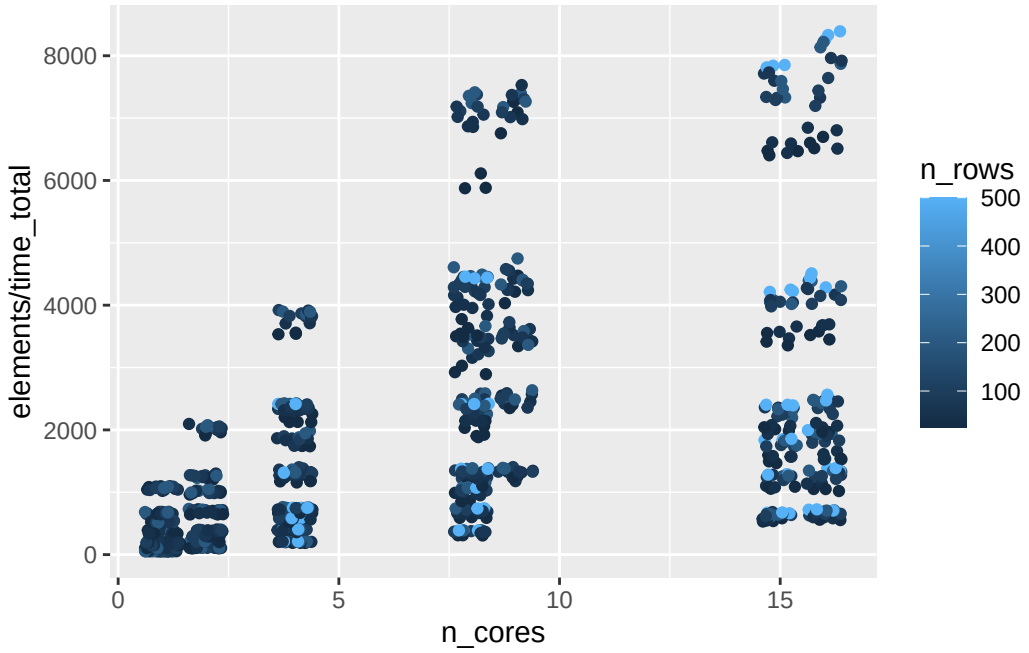
```
ggplot(times_distance, aes(y=elements/time_total, x=n_cores, colour=arch)) +
  geom_jitter()
```




```
ggplot(times_distance, aes(y=elements/time_total, x=n_cores, color=elements)) +
  geom_jitter()
```




```
ggplot(times_distance, aes(y=elements/time_total, x=n_cores, color=n_rows)) +
  geom_jitter()
```



```
ggplot(times_distance, aes(y=elements/time_total, x=n_cores, color=n_cols)) +
  geom_jitter()
```