

VC_Bench_Poster_Figures

Jesse Elder

11/10/2021

A BUNCH OF PREPROCESSING

```
library(tidyverse)

## -- Attaching packages ----- tidyverse 1.3.0 --
## v ggplot2 3.3.3    v purrr  0.3.4
## v tibble  3.1.0    v dplyr  1.0.5
## v tidyr   1.1.3    v stringr 1.4.0
## v readr   1.4.0    v forcats 0.5.1

## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

library(RColorBrewer)
library(reshape2)

##
## Attaching package: 'reshape2'

## The following object is masked from 'package:tidyr':
##
## smiths

library(scales)

##
## Attaching package: 'scales'

## The following object is masked from 'package:purrr':
##
## discard

## The following object is masked from 'package:readr':
##
## col_factor

bench_df<- read.csv("~/Desktop/NoyesLab/All_Benchmarks.csv")
#bench_df

###Dataset Fixing
bench_df$Dataset_num<-ifelse(bench_df$Dataset=="M0.5",0.5,
                             ifelse(bench_df$Dataset=="M1",1,
                                     ifelse(bench_df$Dataset=="M5",5,
```

```

        ifelse(bench_df$Dataset=="M10",10,
              ifelse(bench_df$Dataset=="M15",15,
                    ifelse(bench_df$Dataset=="M25",25,50))))))
bench_df$Dataset_f<-factor(bench_df$Dataset, levels=c("M0.5","M1","M5","M10","M15","M25","M50"))

###Subset Fixing
bench_df$Subset_num<-ifelse(bench_df$Subset=="S5",5,
                           ifelse(bench_df$Subset=="S15",15,
                                   ifelse(bench_df$Subset=="S25",25,35)))
bench_df$Subset<-ifelse(bench_df$Subset=="S5","5 Genomes",
                      ifelse(bench_df$Subset=="S15","15 Genomes",
                              ifelse(bench_df$Subset=="S25","25 Genomes","35 Genomes"))))

bench_df$Subset_f<-factor(bench_df$Subset, levels=c("5 Genomes","15 Genomes","25 Genomes","35 Genomes"))

###Variant Caller Fixing
bench_df$VCaller<-ifelse(bench_df$VCaller=="FB_Out","FreeBayes",
                       ifelse(bench_df$VCaller=="GATK_Out","GATK",
                               ifelse(bench_df$VCaller=="Disco_Out","DiscoSNP++","MetaSNV"))))
bench_df$VCaller_f<-factor(bench_df$VCaller,levels=c("MetaSNV","GATK","FreeBayes","DiscoSNP++"))

cbbPalette<-rev(brewer.pal(n=4,name = "Dark2"))

```

Making color palettes

```

cbbPalette<-rev(brewer.pal(n=4,name = "Dark2"))
VCPalette<-rev(brewer.pal(n=7,name = "Dark2"))
VC_light_palette<-rev(brewer.pal(n=7,name = "BuGn"))

```

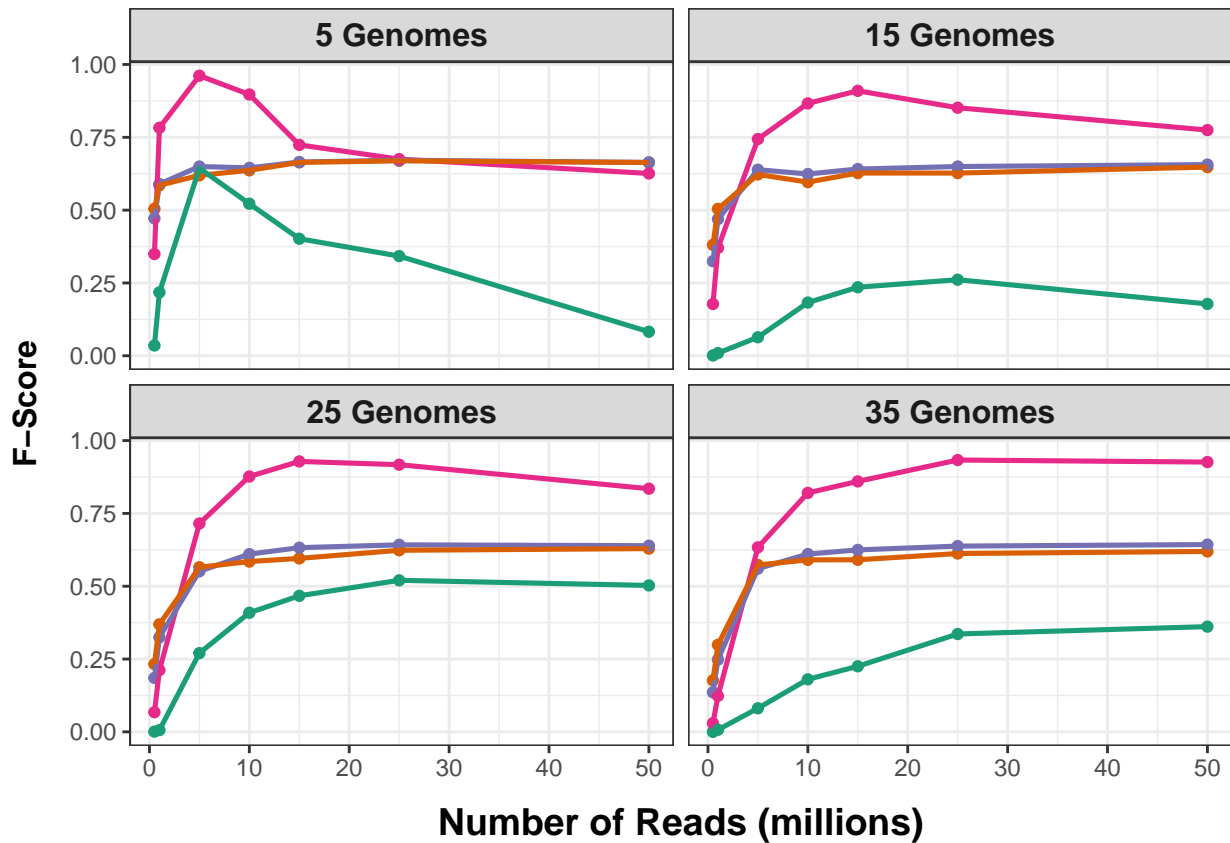
SUBSET AS A FACET

\\

```

###MUST INCLUDE
#1A
ggplot(bench_df,aes(x=Dataset_num,y=F.score,colour=VCaller_f,group=VCaller_f)) +
  geom_point(show.legend = FALSE) + geom_line(size=0.9,show.legend = FALSE) + facet_wrap(~Subset_f,nrow=
  ggtitle("Variant Caller Accuracy by Number of Reads\n and Reference Genomes") +
  xlab("Number of Reads (millions)") + ylab("F-Score") + #labs(col="Variant Caller") +
  scale_color_manual(values=cbbPalette) + theme_bw() +
  theme(plot.title = element_blank(), #plot.title = element_text(hjust=0.5,size=13,face="bold",margin=m
        strip.text.x = element_text(size=12, face="bold"),
        axis.title.x = element_text(size= 14, face="bold",margin=margin(10,0,0,0)),
        axis.title.y = element_text(size=12, face="bold",margin=margin(0,10,0,0)),
        legend.title = element_blank(), legend.text = element_blank())

```

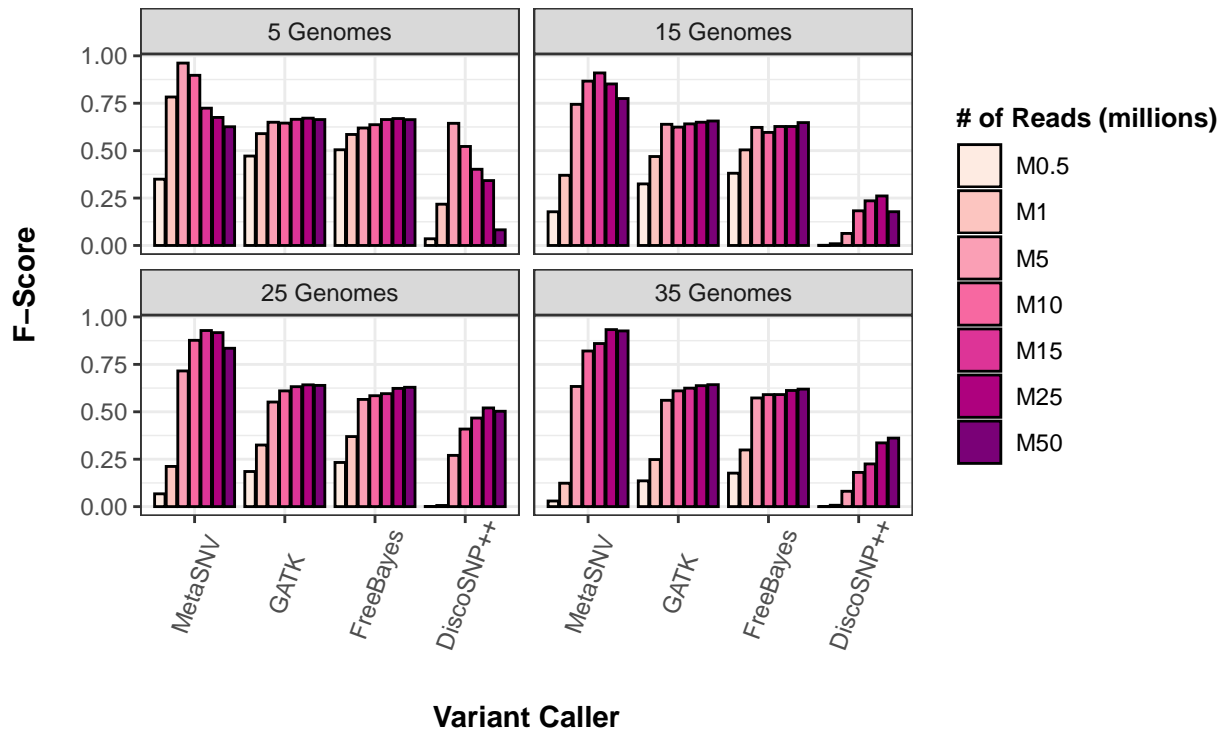


```
#legend.title = element_text(size = 10, face = "bold"), legend.position = "bottom")
###MUST INCLUDE
```

1A

```
###Reasonably good
#1B
ggplot(bench_df, aes(x=VCaller_f, y=F.score, fill=Dataset_f, group=Dataset_f)) +
  geom_col(colour="black", position="dodge2") + facet_wrap(~Subset_f, nrow=2) +
  ggtitle("Variant Caller Accuracy by Number of Reads\n and Reference Genomes") +
  xlab("Variant Caller") + ylab("F-Score") + labs(fill="# of Reads (millions)") +
  scale_fill_brewer(palette = "RdPu") + theme_bw() +
  theme(plot.title = element_text(hjust=0.5, size=13, face="bold", margin=margin(0, 0, 15, 0)),
        axis.title.x = element_text(face="bold", margin=margin(15, 0, 0, 0)),
        axis.text.x = element_text(size=9, angle=70, vjust = 0.6),
        axis.title.y = element_text(face="bold", margin=margin(0, 15, 0, 0)),
        legend.title = element_text(size = 10, face = "bold"))
```

Variant Caller Accuracy by Number of Reads and Reference Genomes



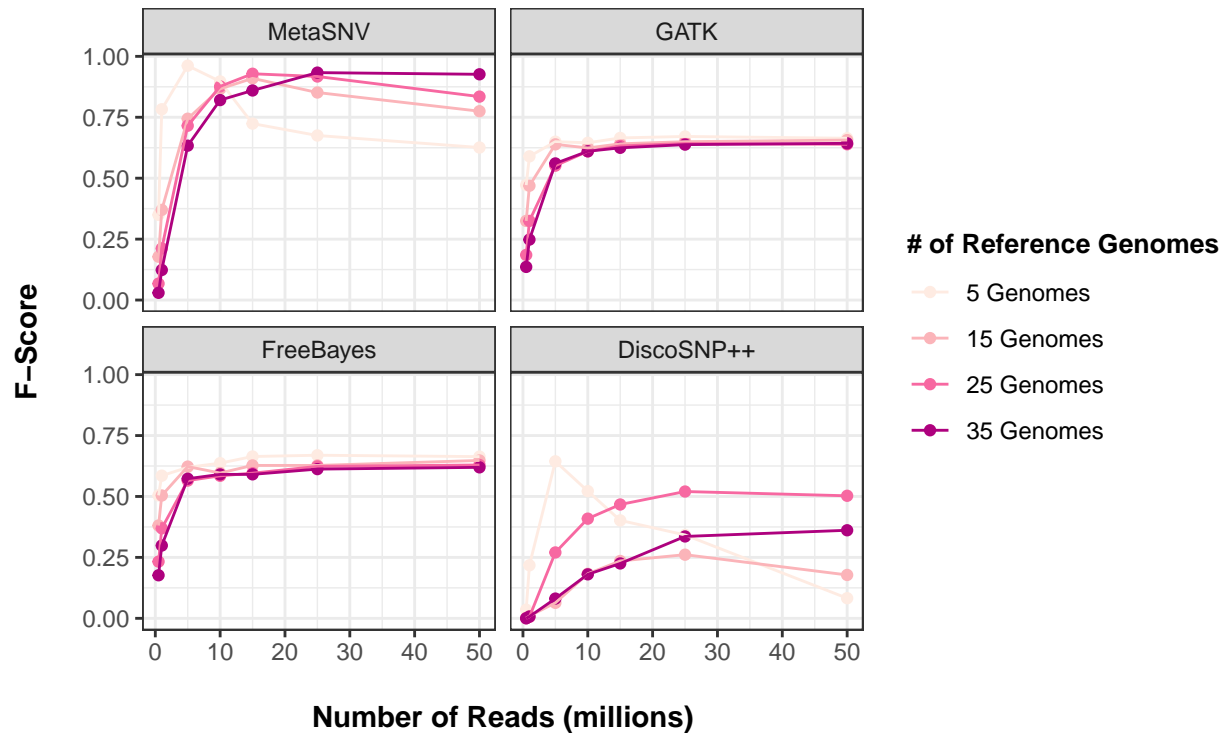
1B

VARIANT CALLER AS A FACET

\\

```
#####CONSIDER USING EITHER LINE GRAPHS
###This one is BETTER
#2A
ggplot(bench_df, aes(x=Dataset_num, y=F.score, colour=Subset_f, group=Subset_f)) +
  geom_point() + geom_line() + facet_wrap(~VCaller_f, nrow=2) +
  ggtitle("Variant Caller Accuracy by Number of Reads\n and Reference Genomes") +
  xlab("Number of Reads (millions)") + ylab("F-Score") + labs(col="# of Reference Genomes") +
  scale_color_brewer(palette = "RdPu") + theme_bw() +
  theme(plot.title = element_text(hjust=0.5, size=13, face="bold", margin=margin(0,0,15,0)),
        axis.title.x = element_text(face="bold", margin=margin(15,0,0,0)),
        axis.title.y = element_text(face="bold", margin=margin(0,15,0,0)),
        legend.title = element_text(size = 10, face = "bold"))
```

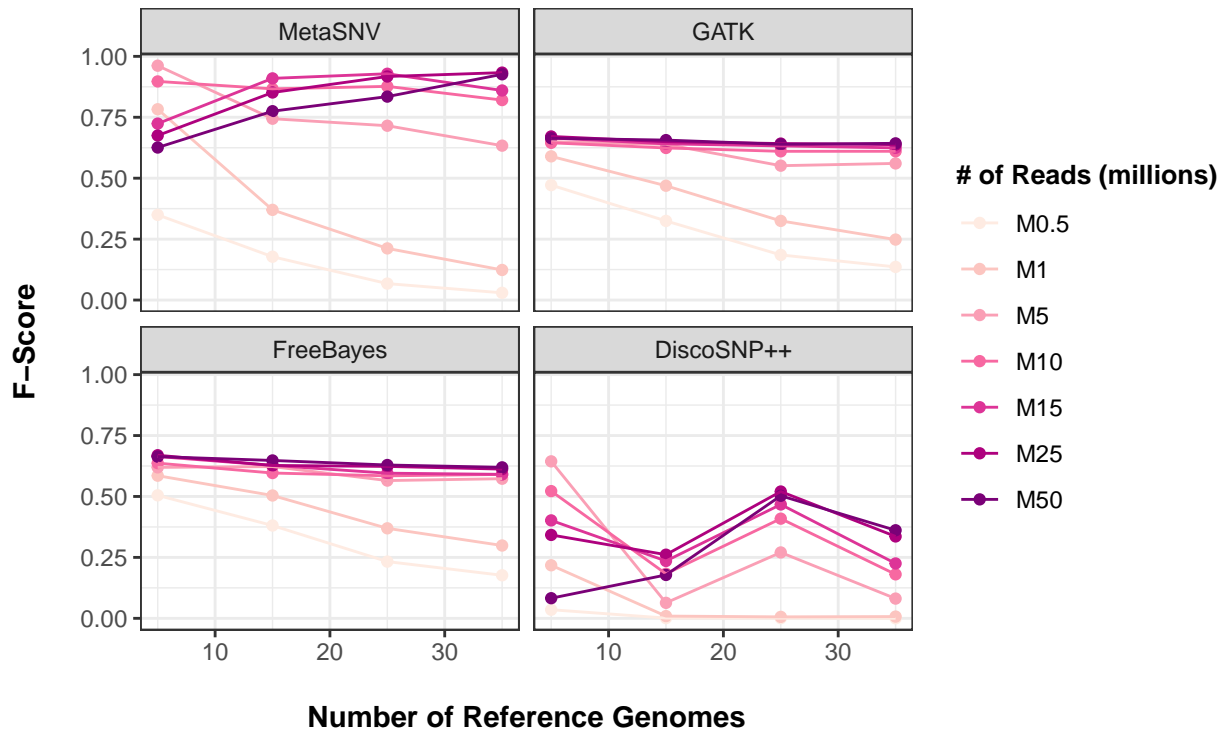
Variant Caller Accuracy by Number of Reads and Reference Genomes



2A

```
###This is also good
#2B
ggplot(bench_df, aes(x=Subset_num, y=F.score, colour=Dataset_f, group=Dataset_f)) +
  geom_point() + geom_line() + facet_wrap(~VCaller_f, nrow=2) +
  ggtitle("Variant Caller Accuracy by Number of Reads\n and Reference Genomes") +
  xlab("Number of Reference Genomes") + ylab("F-Score") + labs(col="# of Reads (millions)") +
  scale_color_brewer(palette = "RdPu") + theme_bw() +
  theme(plot.title = element_text(hjust=0.5, size=13, face="bold", margin=margin(0, 0, 15, 0)),
        axis.title.x = element_text(face="bold", margin=margin(15, 0, 0, 0)),
        axis.title.y = element_text(face="bold", margin=margin(0, 15, 0, 0)),
        legend.title = element_text(size = 10, face = "bold"))
```

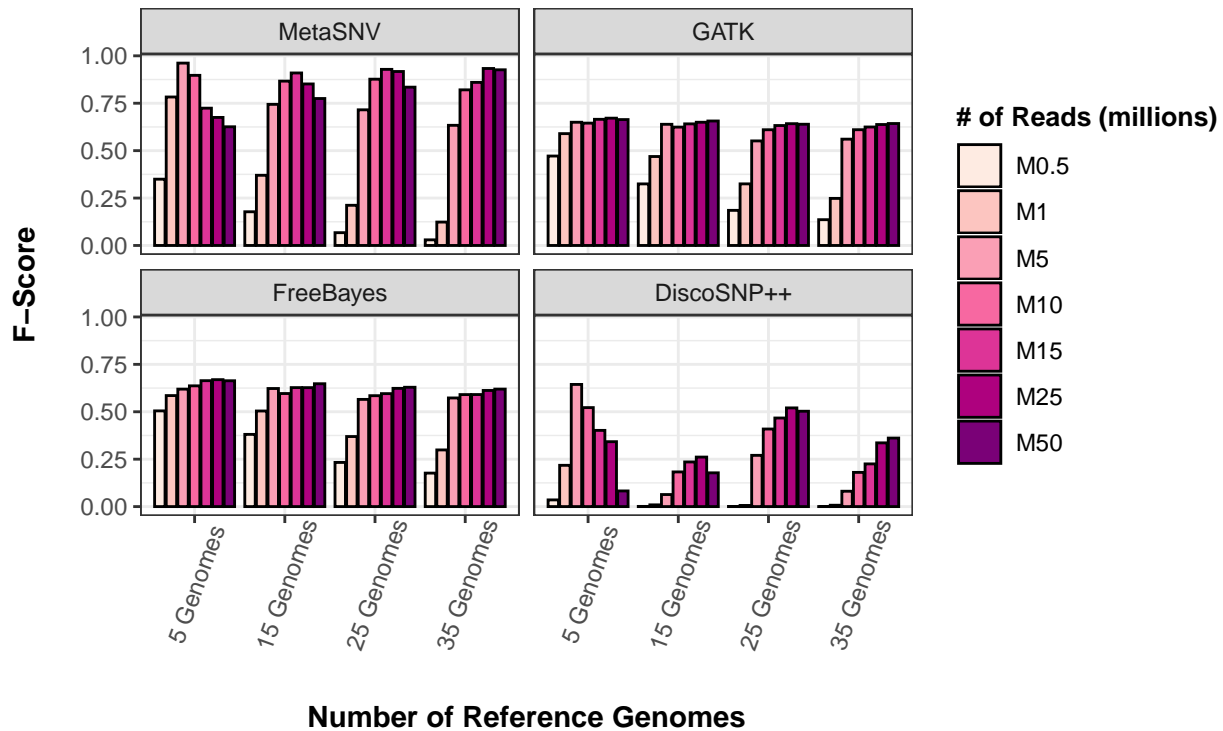
Variant Caller Accuracy by Number of Reads and Reference Genomes



2B

```
#Barplot version is less good but viable
#2C
ggplot(bench_df, aes(x=Subset_f, y=F.score, fill=Dataset_f, group=Dataset_f)) +
  geom_col(colour="black", position="dodge2") + facet_wrap(~VCaller_f, nrow=2) +
  ggtitle("Variant Caller Accuracy by Number of Reads\n and Reference Genomes") +
  xlab("Number of Reference Genomes") + ylab("F-Score") + labs(fill="# of Reads (millions)") +
  scale_fill_brewer(palette = "RdPu") + theme_bw() +
  theme(plot.title = element_text(hjust=0.5, size=13, face="bold", margin=margin(0, 0, 15, 0)),
        axis.title.x = element_text(face="bold", margin=margin(15, 0, 0, 0)),
        axis.text.x = element_text(size=9, angle=70, vjust = 0.6),
        axis.title.y = element_text(face="bold", margin=margin(0, 15, 0, 0)),
        legend.title = element_text(size = 10, face = "bold"))
```

Variant Caller Accuracy by Number of Reads and Reference Genomes



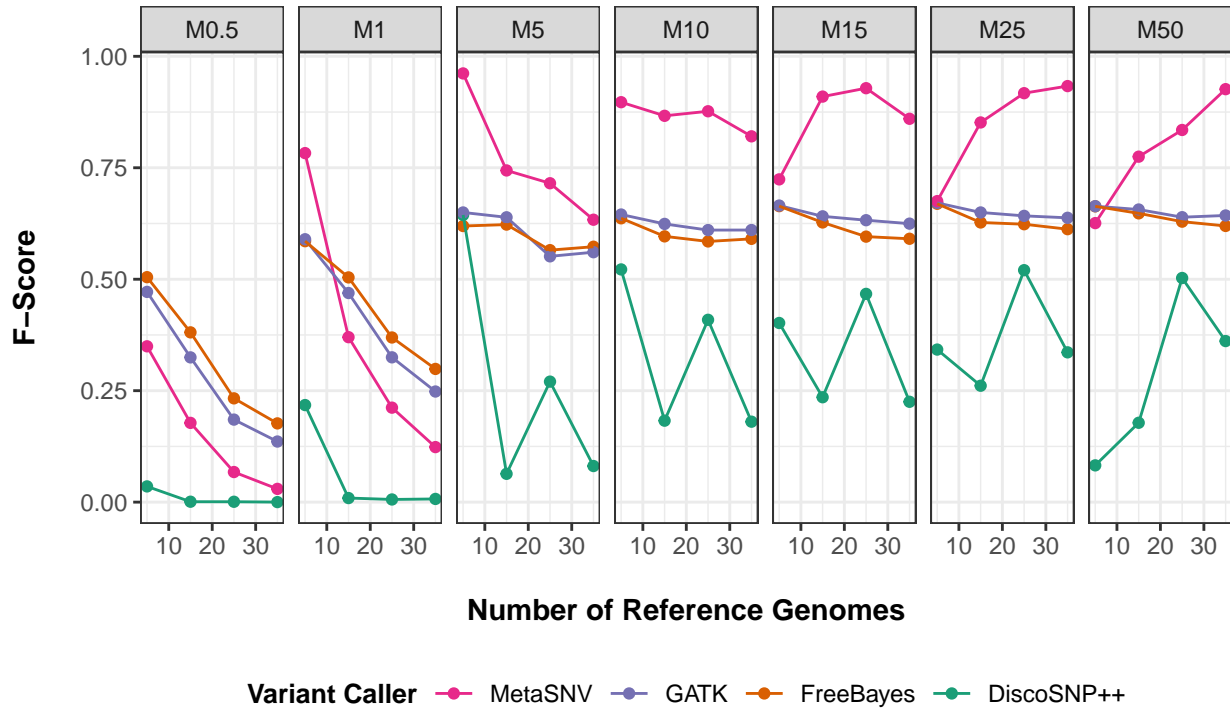
2C

DATASET AS A FACET

\\

```
###INCLUDE THIS ONE IF POSSIBLE -- GOOD STORY HERE
#3A
ggplot(bench_df, aes(x=Subset_num, y=F.score, colour=VCaller_f, group=VCaller_f)) +
  geom_point() + geom_line() + facet_wrap(~Dataset_f, nrow=1) +
  ggtitle("Variant Caller Accuracy by Number of Reads\n and Reference Genomes") +
  xlab("Number of Reference Genomes") + ylab("F-Score") + labs(col="Variant Caller") +
  scale_color_manual(values=cbbPalette) + theme_bw() +
  theme(plot.title = element_text(hjust=0.5, size=13, face="bold", margin=margin(0, 0, 15, 0)),
        axis.title.x = element_text(face="bold", margin=margin(15, 0, 0, 0)),
        axis.title.y = element_text(face="bold", margin=margin(0, 15, 0, 0)),
        legend.title = element_text(size = 10, face = "bold"), legend.position = "bottom")
```

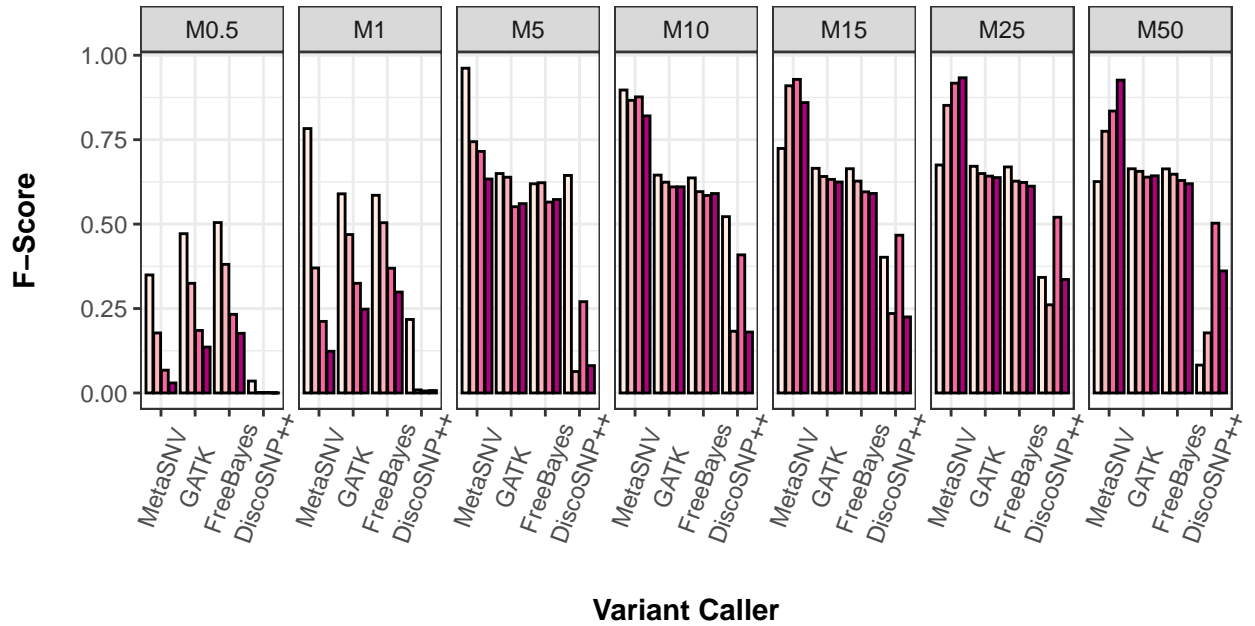
Variant Caller Accuracy by Number of Reads and Reference Genomes



3A

```
#3B
ggplot(bench_df, aes(x=VCaller_f, y=F.score, fill=Subset_f, group=Subset_f)) +
  geom_col(colour="black", position="dodge2") + facet_wrap(~Dataset_f, nrow=1) +
  ggtitle("Variant Caller Accuracy by Number of Reads\n and Reference Genomes") +
  xlab("Variant Caller") + ylab("F-Score") + labs(fill="# of Reference Genomes") +
  scale_fill_brewer(palette = "RdPu") + theme_bw() +
  theme(plot.title = element_text(hjust=0.5, size=13, face="bold", margin=margin(0, 0, 15, 0)),
        axis.title.x = element_text(face="bold", margin=margin(15, 0, 0, 0)),
        axis.text.x = element_text(size=9, angle=70, vjust = 0.6),
        axis.title.y = element_text(face="bold", margin=margin(0, 15, 0, 0)),
        legend.title = element_text(size = 10, face = "bold"), legend.position = "bottom")
```


Variant Caller Accuracy by Number of Reads and Reference Genomes



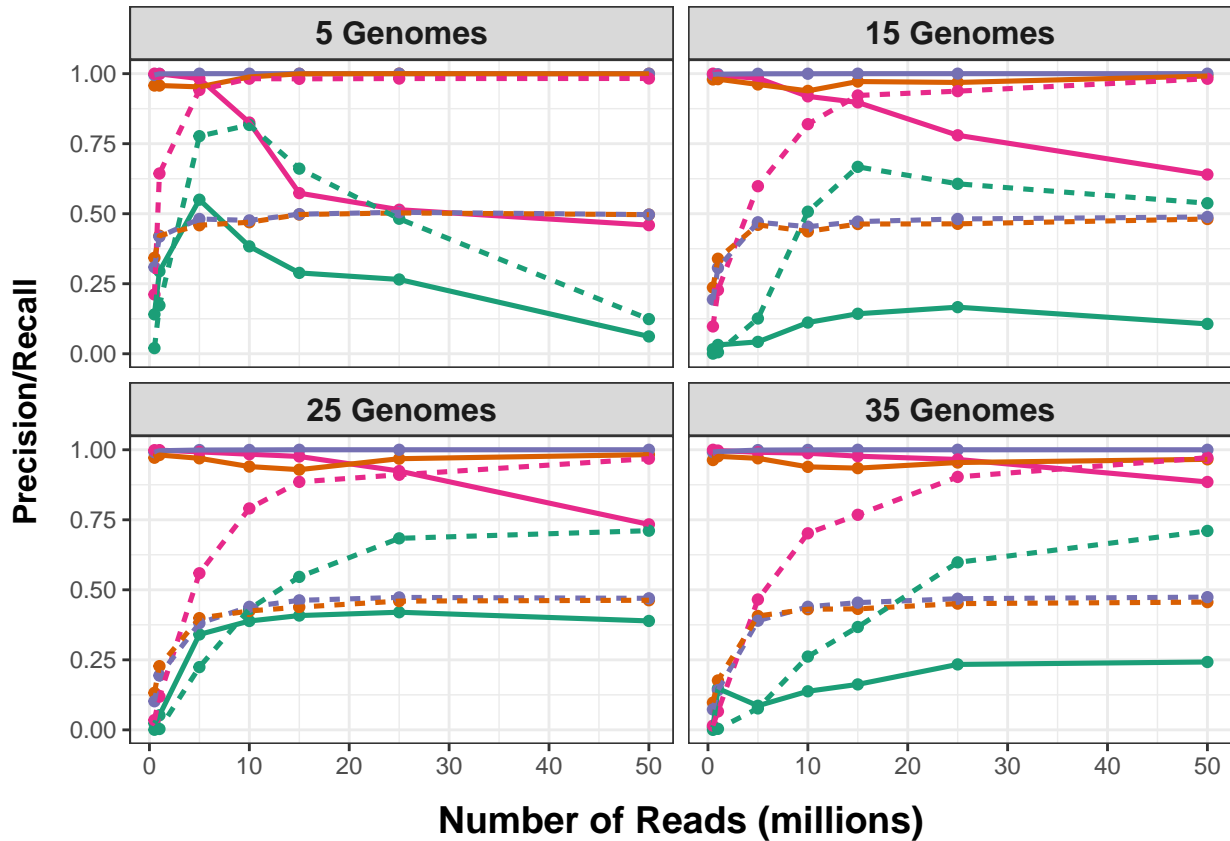
of Reference Genomes 5 Genomes 15 Genomes 25 Genomes 35 Genomes

3B

Figures using Precision/Recall

```
bench_melt<-gather(bench_df,"PR","PR_value",7:8)

ggplot(bench_melt,aes(x=Dataset_num,y=PR_value,colour=VCaller_f,group=interaction(VCaller_f,PR))) +
  geom_point(show.legend = FALSE) + geom_line(show.legend = FALSE,size=0.9,aes(linetype=PR)) + facet_wrap(
  ggtitle("Variant Caller Accuracy by Number of Reads\n and Reference Genomes") +
  xlab("Number of Reads (millions)") + ylab("Precision/Recall") + labs(col="Variant Caller") +
  scale_color_manual(values=cbbPalette) + theme_bw() +
  theme(plot.title = element_blank(), #plot.title = element_text(hjust=0.5,size=13,face="bold",margin=m
    strip.text.x = element_text(size=12, face="bold"),
    axis.title.x = element_text(size= 14, face="bold",margin=margin(10,0,0,0)),
    axis.title.y = element_text(size=12, face="bold",margin=margin(0,10,0,0)),
    legend.title = element_blank(), legend.text = element_blank())
```

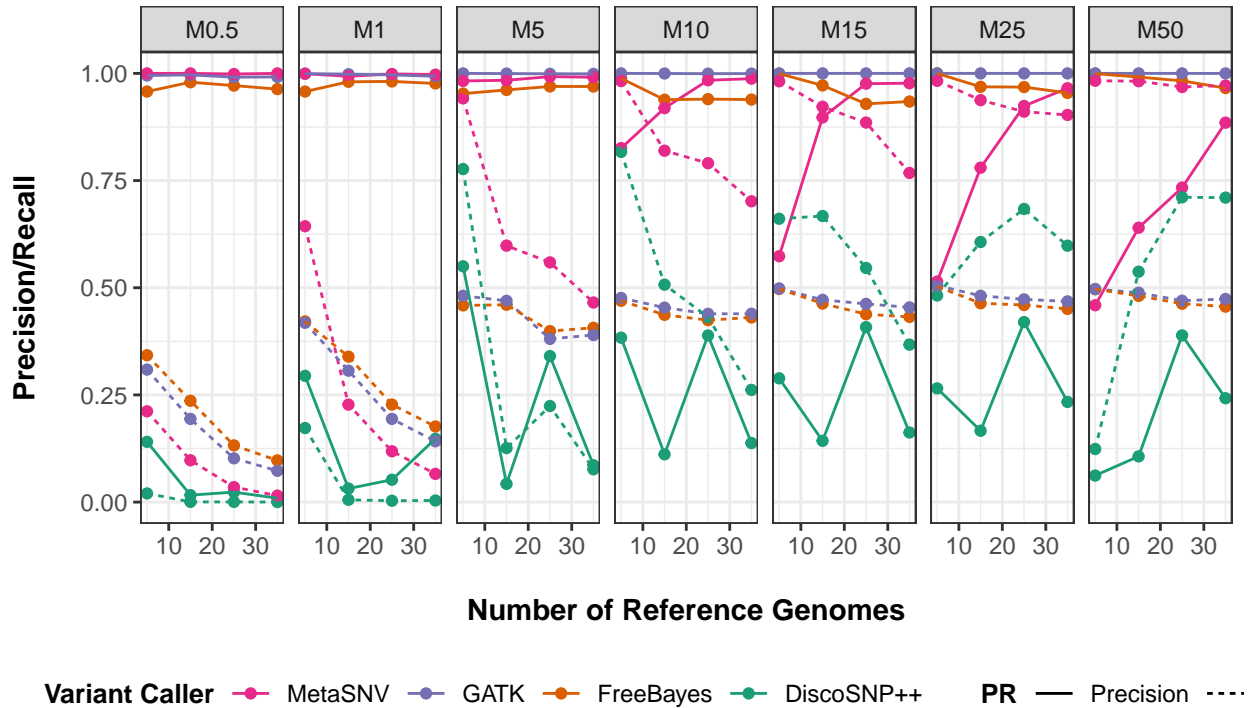


```
#legend.title = element_text(size = 10, face = "bold"), legend.position = "bottom",
#legend.box = "vertical", legend.margin=margin(t=-0.1, unit="cm"))
```

1PR

```
ggplot(bench_melt, aes(x=Subset_num, y=PR_value, colour=VCaller_f, group=interaction(VCaller_f, PR))) +
  geom_point() + geom_line(aes(linetype=PR)) + facet_wrap(~Dataset_f, nrow=1) +
  ggtitle("Variant Caller Accuracy by Number of Reads\n and Reference Genomes") +
  xlab("Number of Reference Genomes") + ylab("Precision/Recall") + labs(col="Variant Caller") +
  scale_color_manual(values=cbbPalette) + theme_bw() +
  theme(plot.title = element_text(hjust=0.5, size=13, face="bold", margin=margin(0,0,15,0)),
        axis.title.x = element_text(face="bold", margin=margin(15,0,0,0)),
        axis.title.y = element_text(face="bold", margin=margin(0,15,0,0)),
        legend.title = element_text(size = 10, face = "bold"), legend.position = "bottom") ##
```

Variant Caller Accuracy by Number of Reads and Reference Genomes



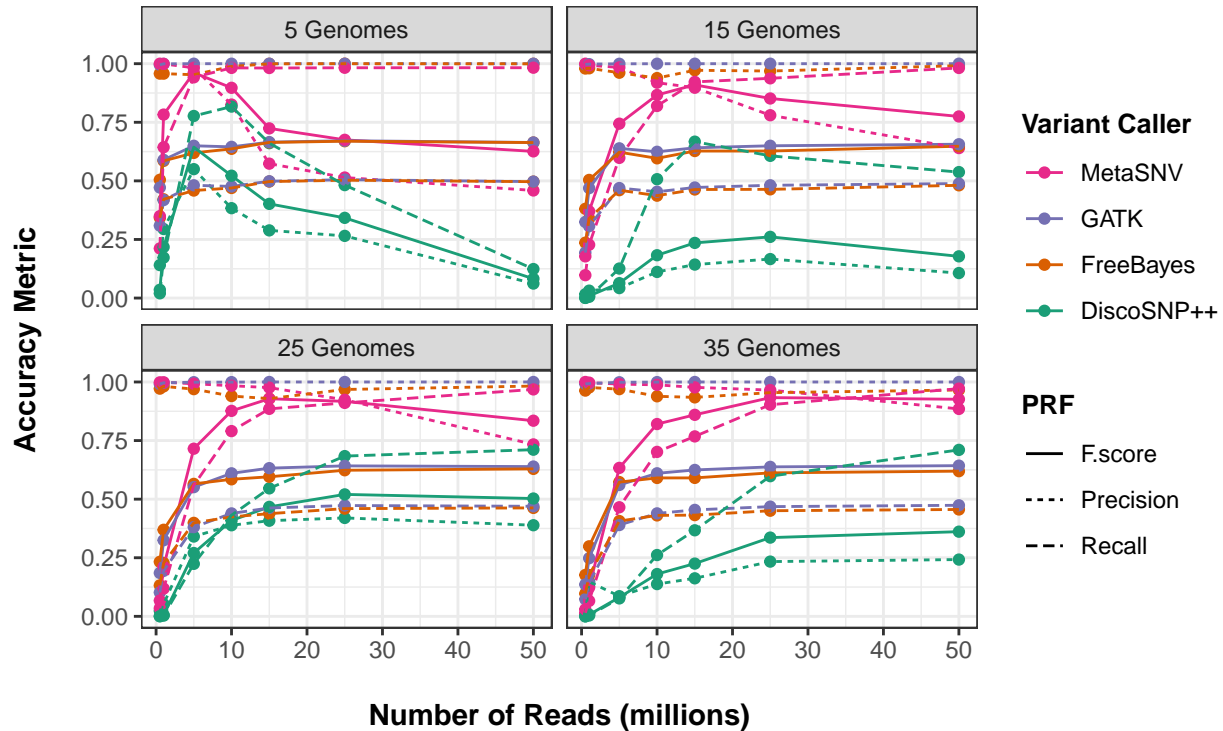
```
#guides(fill=guide_legend(nrow=2,byrow=TRUE))
```

3PR

```
bench_super_melt<-gather(bench_df, "PRF", "PRF_value", 7:9)

ggplot(bench_super_melt, aes(x=Dataset_num, y=PRF_value, colour=VCaller_f, group=interaction(VCaller_f, PRF))) +
  geom_point() + geom_line(aes(linetype=PRF)) + facet_wrap(~Subset_f, nrow=2) +
  ggtitle("Variant Caller Accuracy by Number of Reads\n and Reference Genomes") +
  xlab("Number of Reads (millions)") + ylab("Accuracy Metric") + labs(col="Variant Caller") +
  scale_color_manual(values=cbbPalette) + theme_bw() +
  theme(plot.title = element_text(hjust=0.5, size=13, face="bold", margin=margin(0, 0, 15, 0)),
        axis.title.x = element_text(face="bold", margin=margin(15, 0, 0, 0)),
        axis.title.y = element_text(face="bold", margin=margin(0, 15, 0, 0)),
        legend.title = element_text(size = 10, face = "bold") )
```

Variant Caller Accuracy by Number of Reads and Reference Genomes



1PRF

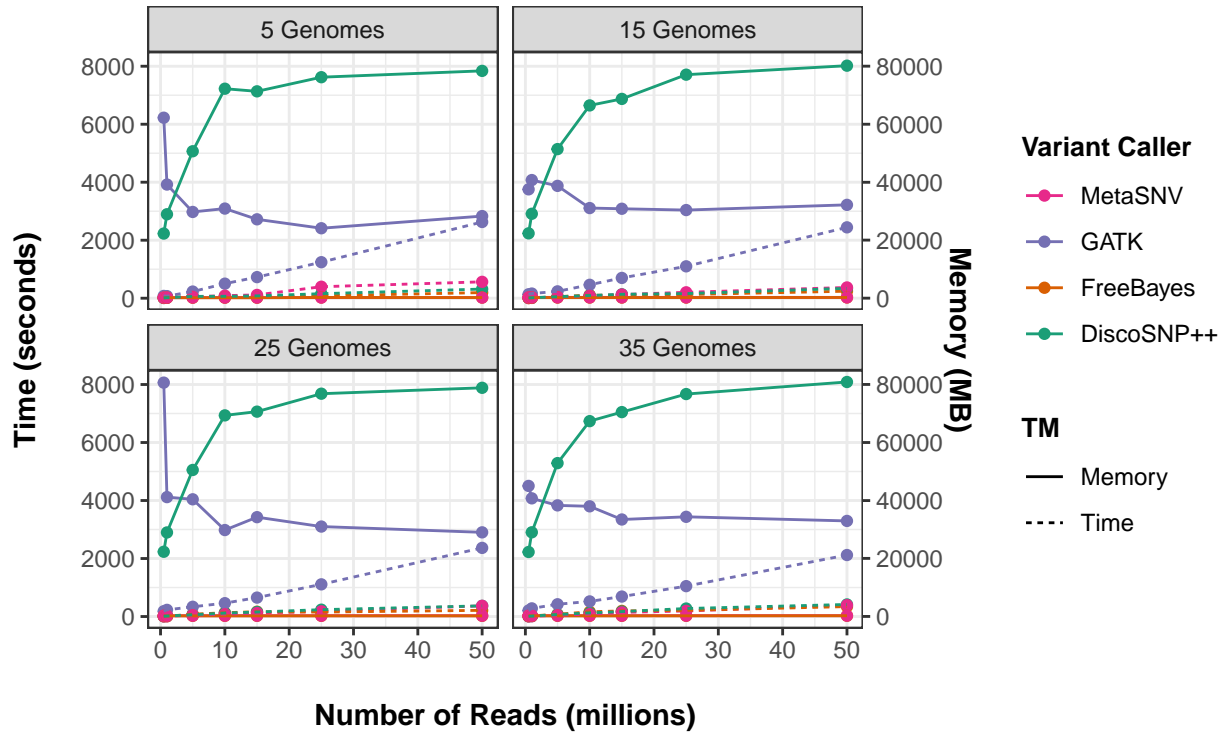
Time and Memory Benchmarks

These ones aren't that good

```
tm_df<-gather(bench_df,"TM","TM_value",10:11)

#Scaled axis, unscaled data
ggplot(tm_df,aes(x=Dataset_num,y=TM_value,colour=VCaller_f,group=interaction(VCaller_f,TM))) +
  geom_point() + geom_line(aes(linetype=TM)) + facet_wrap(~Subset_f,nrow=2) +
  ggtitle("Time & Memory Benchmarks by Number of Reads\n and Reference Genomes") +
  xlab("Number of Reads (millions)") + ylab("Time (seconds)") + labs(col="Variant Caller") +
  scale_y_continuous(name="Time (seconds)", sec.axis = sec_axis(~ 10*., name="Memory (MB)")) +
  scale_color_manual(values=cbbPalette) + theme_bw() +
  theme(plot.title = element_text(hjust=0.5,size=13,face="bold",margin=margin(0,0,15,0)),
        axis.title.x = element_text(face="bold",margin=margin(15,0,0,0)),
        axis.title.y = element_text(face="bold",margin=margin(0,15,0,0)),
        legend.title = element_text(size = 10, face = "bold") )
```

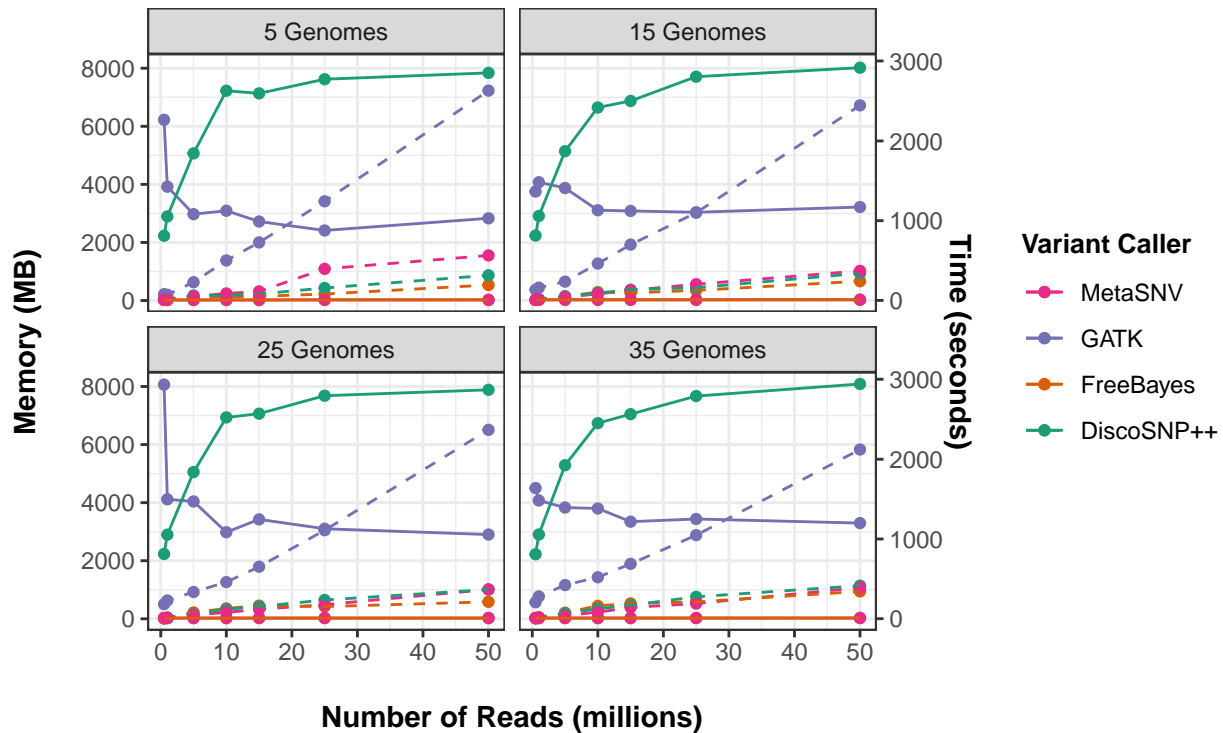
Time & Memory Benchmarks by Number of Reads and Reference Genomes



TM1

```
#Scaled axis and data
ggplot(bench_df) +
  geom_point(mapping = aes(x=Dataset_num,y=Time*2.75,colour=VCaller_f)) +
  geom_line(mapping = aes(x=Dataset_num,y=Time*2.75,colour=VCaller_f),linetype=2) +
  geom_point(mapping = aes(x=Dataset_num,y=Memory,colour=VCaller_f)) +
  geom_line(mapping = aes(x=Dataset_num,y=Memory,colour=VCaller_f),linetype=1) +
  facet_wrap(~Subset_f,nrow=2) +
  ggtitle("Time & Memory Benchmarks by Number of Reads\n and Reference Genomes") +
  xlab("Number of Reads (millions)") + ylab("Time (seconds)") + labs(col="Variant Caller") +
  scale_y_continuous(name="Memory (MB)", sec.axis = sec_axis(~ ./2.75, name="Time (seconds)")) +
  scale_color_manual(values=cbbPalette) + theme_bw() +
  theme(plot.title = element_text(hjust=0.5,size=13,face="bold",margin=margin(0,0,15,0)),
        axis.title.x = element_text(face="bold",margin=margin(15,0,0,0)),
        axis.title.y = element_text(face="bold",margin=margin(0,15,0,0)),
        legend.title = element_text(size = 10, face = "bold") )
```

Time & Memory Benchmarks by Number of Reads and Reference Genomes

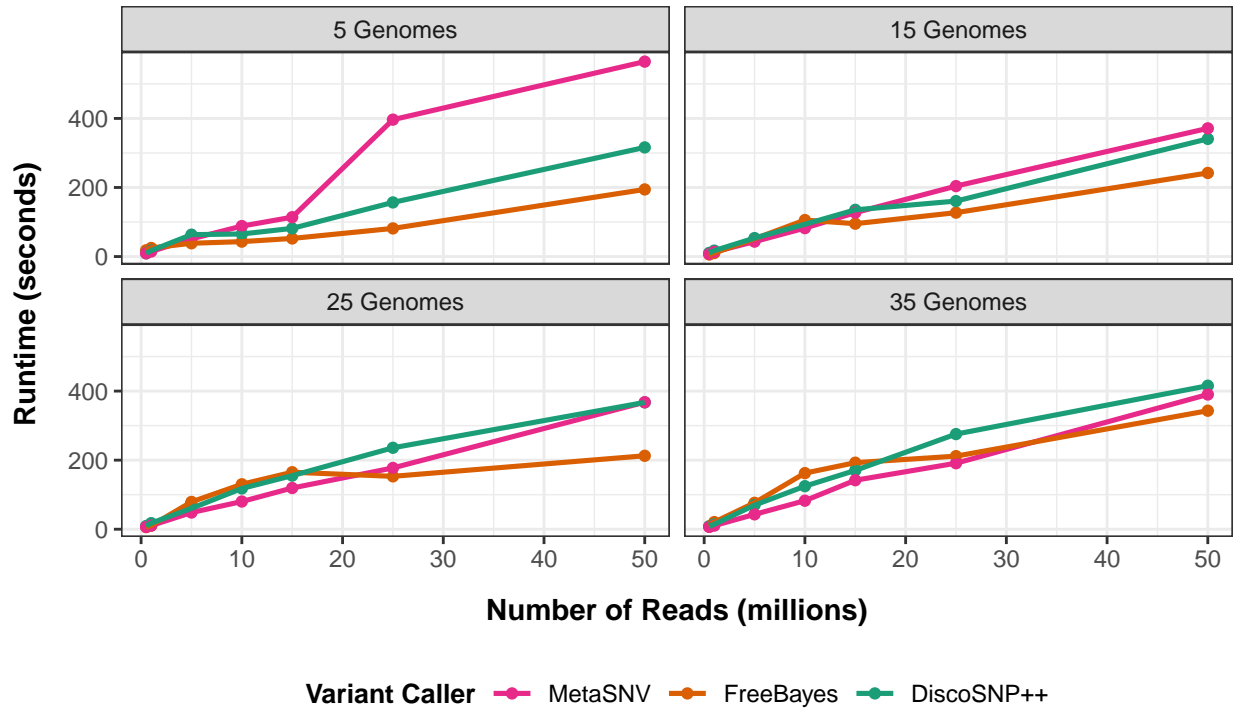


TM2

These are Fab

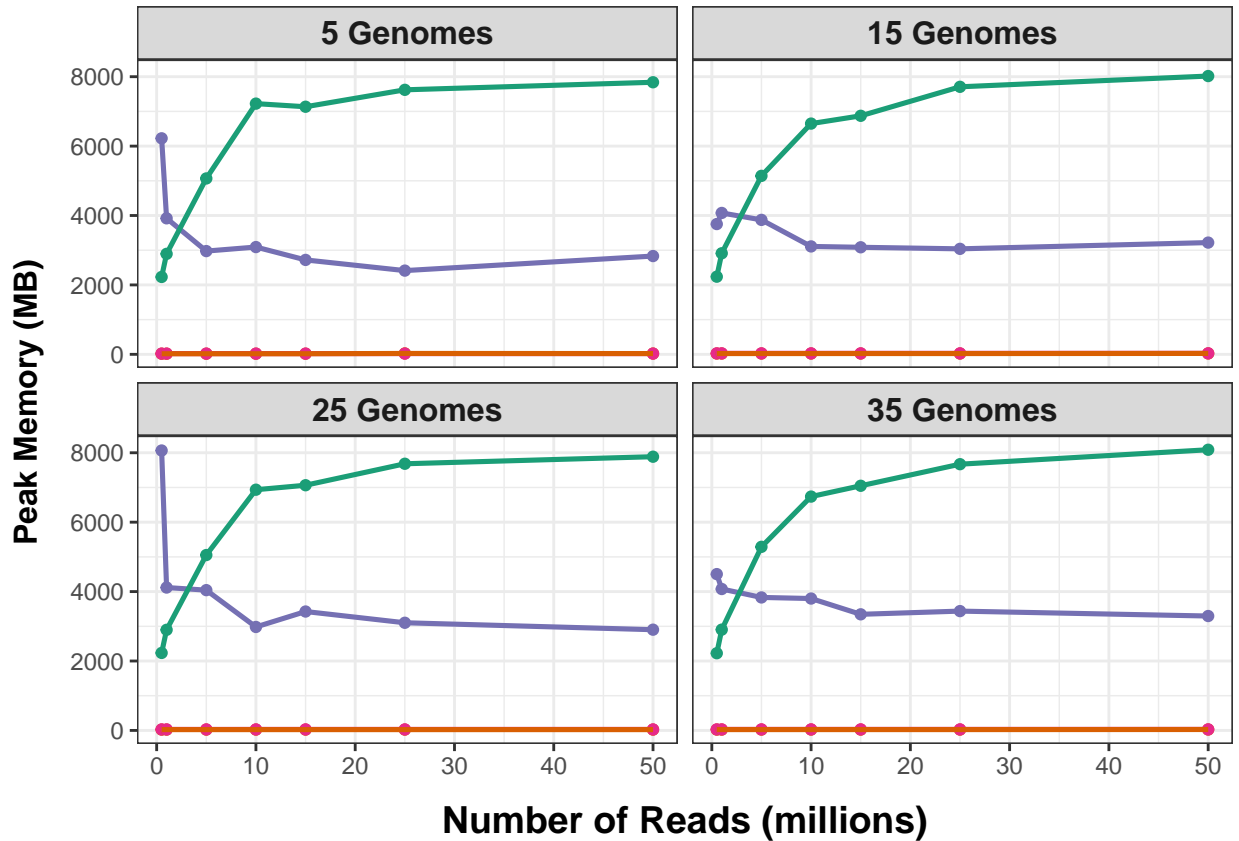
```
ggplot(bench_df[bench_df$VCaller!="GATK",],aes(x=Dataset_num,y=Time,colour=VCaller_f,group=VCaller_f)) +
  geom_point() + geom_line(size=0.9) + facet_wrap(~Subset_f,nrow=2) +
  ggtitle("Runtime by Number of Reads\n and Reference Genomes") +
  xlab("Number of Reads (millions)") + ylab("Runtime (seconds)") + labs(col="Variant Caller") +
  scale_color_manual(values=cbbPalette[c(1,3,4)]) + theme_bw() +
  theme(plot.title = element_text(hjust=0.5,size=13,face="bold",margin=margin(0,0,15,0)),
        axis.title.x = element_text(face="bold",margin=margin(10,0,0,0)),
        axis.title.y = element_text(face="bold",margin=margin(0,10,0,0)),
        legend.title = element_text(size = 10, face = "bold"),legend.position = "bottom")
```

Runtime by Number of Reads and Reference Genomes



T1

```
ggplot(bench_df, aes(x=Dataset_num, y=Memory, colour=VCaller_f, group=VCaller_f)) +
  geom_point(show.legend = FALSE) + geom_line(show.legend = FALSE, size=0.9) + facet_wrap(~Subset_f, nrow=1)
ggtitle("Peak Memory by Number of Reads\n and Reference Genomes") +
  xlab("Number of Reads (millions)") + ylab("Peak Memory (MB)") + labs(col="Variant Caller") +
  scale_color_manual(values=cbbPalette) + theme_bw() +
  theme(plot.title = element_blank(), #plot.title = element_text(hjust=0.5, size=13, face="bold", margin=m
        strip.text.x = element_text(size=12, face="bold"),
        axis.title.x = element_text(size= 14, face="bold", margin=margin(10,0,0,0)),
        axis.title.y = element_text(size=12, face="bold", margin=margin(0,10,0,0)),
        legend.title = element_blank(), legend.text = element_blank())
```



```
#legend.title = element_text(size = 10, face = "bold"), legend.position = "bottom")
```

M1

Genome Heatmap

```
five<-read.csv("~/Desktop/NoyesLab/SNPSubs/SNPLog_subset5.csv")
fifteen<-read.csv("~/Desktop/NoyesLab/SNPSubs/SNPLog_subset15.csv")
twentyfive<-read.csv("~/Desktop/NoyesLab/SNPSubs/SNPLog_subset25.csv")
thirtyfive<-read.csv("~/Desktop/NoyesLab/SNPSubs/SNPLog_subset35.csv")

#Heat_df<-data.frame(Five=unique(five$CHROM),Fifteen=unique(fifteen$CHROM),TwentyFive=unique(fifteen$CHROM),ThirtyFive=unique(thirtyfive$CHROM))

Heat_df<-data.frame(Full=unique(thirtyfive$CHROM))

Heat_df$5 Genomes<-ifelse(Heat_df$Full %in% unique(five$CHROM),1,0)
Heat_df$15 Genomes<-ifelse(Heat_df$Full %in% unique(fifteen$CHROM),1,0)
Heat_df$25 Genomes<-ifelse(Heat_df$Full %in% unique(twentyfive$CHROM),1,0)
Heat_df$35 Genomes<-ifelse(Heat_df$Full %in% unique(thirtyfive$CHROM),1,0)

Heat_gather<-gather(Heat_df,"Genome_num","Genome_ext",2:5)

Heat_gather$Genome_num_f<-factor(Heat_gather$Genome_num,levels=c("35 Genomes","25 Genomes","15 Genomes","5 Genomes"))
```



```
ggplot(Heat_gather,aes(x=Full,y=Genome_num_f)) + geom_tile(col="black",aes(fill=factor(Genome_ext))) +
  scale_fill_manual(labels=c("Excluded", "Included"),values=c("0"="light grey","1"="red3")) + labs(fill=
  xlab("Reference Genomes") + ylab("Subset of Genomes") +
  theme(plot.title = element_text(hjust=0.5,size=13,face="bold",margin=margin(0,0,15,0)),
        axis.title.x = element_text(size=12,face="bold",margin=margin(10,0,0,0)),
        axis.text.x = element_text(size=9,angle=70,vjust = 0.6),
        axis.title.y = element_text(size=12,face="bold",margin=margin(0,10,0,0)),
        legend.title = element_text(size = 12, face = "bold"), legend.text = element_text(size=12), leg
```

