Supplementary Figures

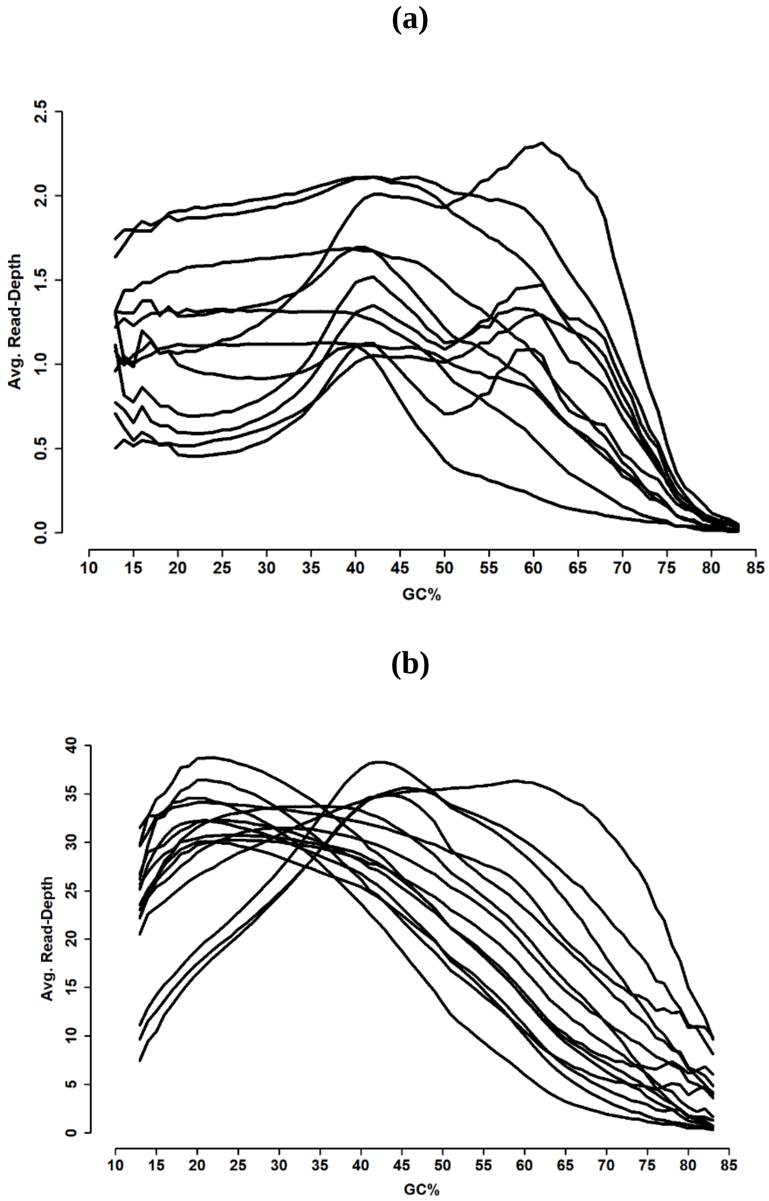

Figure S1. Sample-specific variations in read depth (calculated from VCF files) against different GC%. Each line represent one animal. (a) 12 animals with shallow sequence coverage (average coverage ≤2x); (b) 15 animals with deep sequence coverage (average coverage ≥30x).

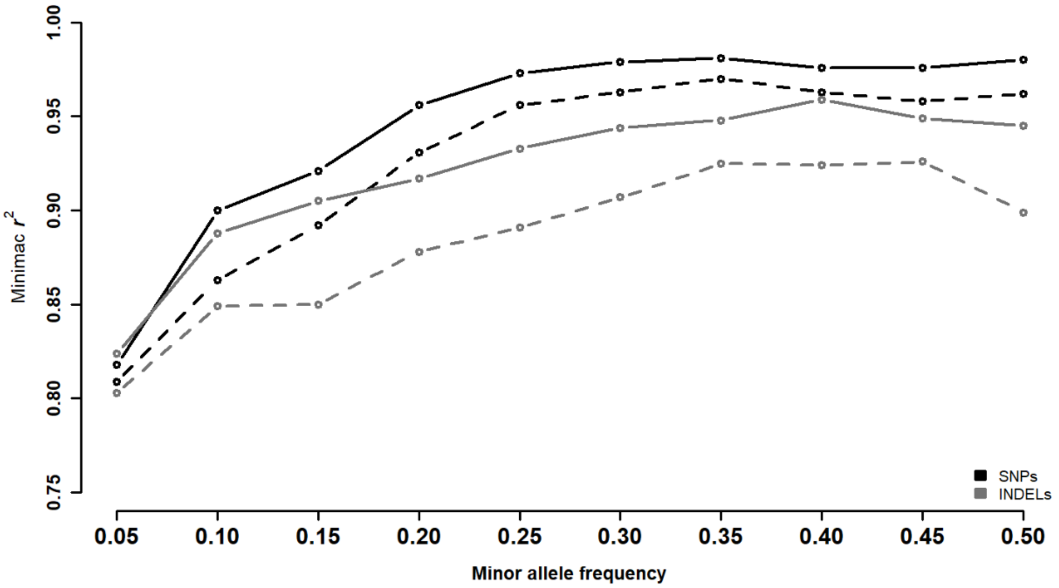

Figure S2. Average downstream imputation accuracy for SNPs and indels on chromosome 29. For each MAF bin of 5%, the average of Minimac r^2 was calculated. Dashed lines: reference panel phased using Beagle ($\text{Ref}_{\text{Beagle}}$). Solid lines: reference panel phased using Beagle followed by re-phasing using SHAPEIT ($\text{Ref}_{\text{Beagle\_SHAPEIT}}$).

Supplementary Tables

Table S1. Percentage of WGS genotypes and markers with Beagle GP <0.99

Chromosome	WGS markers	Total genotypes (GP <0.99)	Markers where >10% genotypes had GP <0.99		
			SNPs	Indels	Deletions
Chr1	922,721	2.2%	6.0%	16.5%	63.9%
Chr2	760,671	2.3%	6.1%	16.4%	64.5%
Chr3	683,688	2.5%	6.8%	18.4%	61.6%
Chr4	717,179	2.4%	6.7%	16.7%	66.3%
Chr5	699,826	3.0%	8.1%	19.2%	67.5%
Chr6	723,461	2.3%	6.2%	16.9%	64.9%
Chr7	620,036	2.7%	7.6%	17.9%	65.7%
Chr8	634,425	2.7%	7.5%	18.2%	61.2%
Chr9	613,418	2.5%	6.9%	16.6%	64.6%
Chr10	603,367	2.6%	7.0%	17.5%	65.0%
Chr11	600,604	2.3%	6.1%	17.3%	67.3%
Chr12	628,018	6.0%	16.9%	28.8%	67.8%
Chr13	448,903	2.5%	6.6%	19.2%	69.1%
Chr14	473,492	2.7%	7.6%	17.5%	70.9%
Chr15	532,254	2.6%	7.1%	18.7%	65.9%
Chr16	464,095	2.5%	6.8%	16.7%	67.0%
Chr17	466,624	2.9%	8.1%	19.9%	59.1%
Chr18	380,741	3.2%	8.5%	21.1%	75.9%
Chr19	357,785	3.1%	8.2%	21.7%	82.3%
Chr20	433,584	2.1%	5.5%	15.5%	66.1%
Chr21	416,612	2.6%	7.2%	17.5%	70.6%
Chr22	334,678	2.1%	5.6%	16.7%	61.7%
Chr23	394,583	2.7%	7.7%	19.1%	73.4%
Chr24	382,233	2.1%	5.7%	16.1%	71.1%
Chr25	266,059	2.4%	6.1%	20.2%	75.8%
Chr26	313,408	2.2%	5.8%	18.9%	66.7%
Chr27	297,277	2.6%	7.3%	19.1%	66.4%
Chr28	294,094	2.4%	6.2%	16.9%	66.4%
Chr29	342,261	2.8%	7.5%	19.5%	69.6%

Table S2. Average downstream imputation accuracy for SNPs and indels on chromosome 29

MAF bins	Average Minimac r^2 (SD) for SNPs		Average Minimac r^2 (SD) for INDELs	
	$\text{Ref}_{\text{Beagle}}$	$\text{Ref}_{\text{Beagle\_SHAPEIT}}$	$\text{Ref}_{\text{Beagle}}$	$\text{Ref}_{\text{Beagle\_SHAPEIT}}$
0< MAF ≤5%	0.809 (0.324)	0.818 (0.320)	0.803 (0.301)	0.824 (0.291)
5< MAF ≤10%	0.863 (0.279)	0.900 (0.244)	0.849 (0.254)	0.888 (0.230)
10< MAF ≤15%	0.892 (0.255)	0.921 (0.219)	0.850 (0.253)	0.905 (0.205)
15< MAF ≤20%	0.931 (0.193)	0.956 (0.149)	0.878 (0.238)	0.917 (0.189)
20< MAF ≤25%	0.956 (0.132)	0.973 (0.094)	0.891 (0.194)	0.933 (0.153)
25< MAF ≤30%	0.963 (0.116)	0.979 (0.079)	0.907 (0.181)	0.944 (0.128)
30< MAF ≤35%	0.970 (0.101)	0.981 (0.072)	0.925 (0.167)	0.948 (0.118)
35< MAF ≤40%	0.963 (0.113)	0.976 (0.085)	0.924 (0.173)	0.959 (0.106)
40< MAF ≤45%	0.958 (0.132)	0.976 (0.099)	0.926 (0.172)	0.949 (0.138)
45< MAF <50%	0.962 (0.116)	0.980 (0.076)	0.899 (0.196)	0.945 (0.136)

MAF: Minor allele frequency; 326,838 SNPs and 2,723 indels

Table S3. Average imputation accuracy in 1% MAF bins for SNPs, indels and deletions of the 29 bovine autosomes

MAF bins	**Average Minimac r^2 (SD)**		
	SNPs	**Indels**	**Deletions**
0< MAF ≤1%	0.528 (0.392)	0.550 (0.370)	0.713 (0.343)
2%	0.844 (0.289)	0.828 (0.286)	0.907 (0.177)
3%	0.895 (0.241)	0.879 (0.242)	0.918 (0.169)
4%	0.905 (0.231)	0.893 (0.225)	0.943 (0.133)
5%	0.910 (0.227)	0.896 (0.219)	0.967 (0.068)
6%	0.915 (0.220)	0.900 (0.211)	0.950 (0.133)
7%	0.915 (0.221)	0.904 (0.206)	0.960 (0.132)
8%	0.919 (0.217)	0.898 (0.212)	0.959 (0.130)
9%	0.923 (0.212)	0.904 (0.208)	0.960 (0.114)
10%	0.924 (0.213)	0.899 (0.212)	0.944 (0.171)
11%	0.929 (0.205)	0.913 (0.195)	0.966 (0.113)
12%	0.932 (0.202)	0.912 (0.192)	0.960 (0.086)
13%	0.939 (0.190)	0.916 (0.187)	0.962 (0.118)
14%	0.941 (0.186)	0.924 (0.174)	0.938 (0.192)
15%	0.948 (0.173)	0.921 (0.183)	0.972 (0.076)
16%	0.952 (0.166)	0.926 (0.174)	0.965 (0.128)
17%	0.955 (0.158)	0.929 (0.172)	0.968 (0.123)
18%	0.960 (0.148)	0.931 (0.170)	0.930 (0.211)
19%	0.963 (0.142)	0.938 (0.154)	0.956 (0.152)
20%	0.965 (0.136)	0.935 (0.160)	0.972 (0.110)
21%	0.966 (0.136)	0.943 (0.149)	0.935 (0.197)
22%	0.968 (0.129)	0.940 (0.153)	0.988 (0.023)
23%	0.968 (0.129)	0.938 (0.153)	0.961 (0.125)
24%	0.970 (0.125)	0.943 (0.147)	0.929 (0.225)

25%	0.973 (0.120)	0.943 (0.159)	0.931 (0.216)
26%	0.972 (0.122)	0.941 (0.158)	0.960 (0.159)
27%	0.974 (0.116)	0.940 (0.156)	0.962 (0.151)
28%	0.974 (0.115)	0.948 (0.136)	0.926 (0.241)
29%	0.974 (0.114)	0.947 (0.142)	0.969 (0.130)
30%	0.975 (0.116)	0.953 (0.131)	0.938 (0.208)
31%	0.975 (0.114)	0.959 (0.111)	0.946 (0.201)
32%	0.976 (0.113)	0.951 (0.137)	0.941 (0.184)
33%	0.976 (0.112)	0.953 (0.136)	0.949 (0.190)
34%	0.976 (0.111)	0.952 (0.143)	0.987 (0.022)
35%	0.977 (0.108)	0.952 (0.148)	0.985 (0.055)
36%	0.977 (0.108)	0.964 (0.109)	0.971 (0.135)
37%	0.979 (0.102)	0.957 (0.127)	0.940 (0.208)
38%	0.980 (0.100)	0.957 (0.129)	0.989 (0.018)
39%	0.979 (0.102)	0.962 (0.113)	0.980 (0.071)
40%	0.980 (0.100)	0.956 (0.128)	0.977 (0.076)
41%	0.981 (0.097)	0.964 (0.111)	0.964 (0.128)
42%	0.982 (0.092)	0.965 (0.109)	0.952 (0.172)
43%	0.980 (0.099)	0.964 (0.102)	0.973 (0.078)
44%	0.980 (0.100)	0.956 (0.124)	0.969 (0.144)
45%	0.980 (0.102)	0.962 (0.115)	0.967 (0.142)
46%	0.980 (0.102)	0.967 (0.113)	0.972 (0.138)
47%	0.979 (0.108)	0.961 (0.125)	0.986 (0.033)
48%	0.980 (0.104)	0.956 (0.126)	0.949 (0.212)
49%	0.981 (0.099)	0.970 (0.092)	0.955 (0.168)
50%	0.979 (0.104)	0.955 (0.141)	0.915 (0.269)

Imputed WGS SNPs = 14,070,960, Indels = 122,054, and deletions = 5,730

Supplementary Methods

Here, we presented the scripts used for calculating expected read depth from the VCF file, Phasing and Imputation. All the scripts are available in github repository https://github.com/MMesbahU/ImputeDelPipeline.git.

1. Expected Read Depth Calculation

1.1 Calculation of GC% in bins of 100 bp

```
# Bovine reference genome UMD3.1 was downloaded from
#"http://128.206.12.216/drupal/sites/bovinegenome.org/files/data/umd3.1/UMD3
.1_chromosomes.fa.gz"
# To Prepare reference FASTA file with 100bp per line
# used "fasta_formatter" from "FASTX Toolkit version 0.0.14"
fasta_formatter -w 100 -i umd31.fa -o formatted_umd31.fa
# FASTA header formating
grep '^>' formatted_umd31.fa | sed -e 's:^>gnl|UMD3.1|::g' \
 -e 's:Chromosome :Chr:g' | awk '{print $1"\t"$2}' > fastaHeader.txt
```

```
# In Python 3.7
```

```
# Calculate GC% in bins of 100 bp
import os
import subprocess
import pandas as pd
# GC_percent function
# output: GC% and Number_of_N_bases
def GC_percent(DNA):
      N_Bases = DNA.count('n') + DNA.count('N')
      GC = float(DNA.count('c') + DNA.count('C') + DNA.count('g') +
DNA.count('G')) * 100.0/(len(DNA)- N_Bases)
      return(round(GC,0), N_Bases)
#### Chromosome dictionary
UMD31_header = pd.read_table('fastaHeader.txt', sep='\t', header=None)
ChromDict = dict( zip(UMD31_header[0], UMD31_header[1]) )
# BED file Header
P1 = subprocess.Popen('echo -e "#Chrom\tSTART\tEND\tGC%\ttot_N_bases" >
umd31_GC_content.bed', shell=True)
os.waitpid(P1.pid, 0)
# Read formatted fasta file
f = open('formatted_umd31.fa', 'r+')
# Append outputs to an existing file
o = open('umd31_GC_content.bed','a+')
lineNum=0
for line in f:
      if line.startswith('>'):
            lineNum=0
            a=line.strip().split(' ')[0]
            b=a.strip().split('|')[2]
            chrom=ChromDict[b]
            continue
      else:
            gc_n=GC_percent(line)
            lineNum += 1
            o.write("\t".join([str(chrom), str((lineNum*100)-99),
str(lineNum*100), str(gc_n[0]), str(gc_n[1]) ]) + "\n")
f.close()
o.close()
# END
```

```
# Filtering the BED interval:
# (1) Keeping Bovine Autosomes: Chr1-Chr29
# (2) Exclude last line from each Chromosome (it is usually < 100bp)
# (3) Exclude intervals that contain "N" bases. These are assembly gaps.
GCfile=umd31_GC_content.bed
for chr in {1..29}
do
    grep -w Chr${chr} ${GCfile} | sed '$ d' | \
      awk -v OFS='\t' '$NF==0 {print $1, $2, $3, $4}' \
      >>  Chr1_29.noGAPs.umd31.bed
done
# (4) Exclude Repeat regions
# repeat annotation:
http://hgdownload.soe.ucsc.edu/goldenPath/bosTau6/database/nestedRepeats.txt
.gz
repeatRegions=nestedRepeats.txt.gz
for chr in {1..29}
do
    zgrep -w chr${chr} ${repeatRegions} | \
        awk -v OFS='\t' '{gsub(/chr/, "Chr",$2)}{print $2, $3, $4}'|\
        sort -V | uniq >>  Chr1_29_NestedRepeats.bed
done
# bedtools intersect
intersectBed -a Chr1_29.noGAPs.umd31.bed -b Chr1_29_NestedRepeats.bed \
    -v > Chr1_29.noGAPs.noRepeats.umd31.bed
## (5) Exclude known CNV and SV regions, such as from DGVA -
# studies with UMD3.1 mapping: such as,
# estd223_Boussaha_et_al_2015; nstd135_Karimi_et_al_2016
# nstd131_Keel_et_al_2016; estd234_Mesbah-Uddin_et_al_2017
# nstd69_Bickhart_et_al_2012; nstd60_Hou_et_al_2011
# nstd61_Hou_et_al_2011b; nstd56_Liu_et_al_2010
wget -i CNV_SV_file_from_DGVA_22Jan2019.list
# Keeping CNVs or SV region <= 1MB size, in authosomes
for file in $(ls *Bos_taurus_UMD_3.1*); do
zgrep -v '^#' ${file}|awk -v OFS='\t' '{print $1,$4,$5,($5-$4+1)}'| sed -e
's:chr::g' -e 's:Chr::g' | awk awk -v OFS='\t' '$1~/^[0-9*]/ && $4<=1e6
{print "Chr"$1, $2, $3}' | sort -V | uniq >>
Chr1_29.UMD31_DGVA_CNVs_SVs.bed; done
# Excluding CNV/SV regions
intersectBed \
      -a Chr1_29.noGAPs.noRepeats.umd31.bed \
      -b Chr1_29.UMD31_DGVA_CNVs_SVs.bed \
      -v >> Chr1_29.noGAPs.noRepeats.noCNVs.umd31.bed
```

1.2 Calculation of genome-wide average read depth for each animal for each 1% GC bins

```
GC_annotation=Chr1_29.noGAPs.noRepeats.noCNVs.umd31.bed
# Extract Read depth from VCF file
# Filter: Bi-allelic SNPs only; SNP quality >=30;
# no SNPs within 10 bp of one other
for chr in {1..29}
do
vcftools --gzvcf Chr${chr}.VCF_1KBGP.vcf.gz \
       --minQ 30 \
       --min-alleles 2 \
       --max-alleles 2 \
       --remove-indels \
       --thin 10 \
       --keep IDs_for_597_animals.list \
       --extract-FORMAT-info AD \
       --out temp.Chr${chr}.readDepth
#
cat temp.Chr${chr}.readDepth.AD.FORMAT | perl -ane 'if ($.==1) {print
join("\t",@F),"\n"; next} print shift @F,"\t",shift @F;while(@F){
$v=shift @F;($v1,$v2)=split(",",$v); print "\t",$v1+$v2} print "\n";' >
Output_Read_depth_Chr${chr}.txt
#
totalCols=`awk 'NR==2 {print NF}' Output_Read_depth_Chr${chr}.txt`
# BED style
paste -d '\t' <(awk -v OFS='\t' 'NR>1 {print $1, $2-1, $2}'
Output_Read_depth_Chr${chr}.txt) <(awk -v OFS='\t' 'NR>1 {print $0}'
Output_Read_depth_Chr${chr}.txt | cut -f3-${totalCols} ) >>
Output_Read_depth_Chr${chr}.bed
# Bedtools intersect
while read lines
do
intersectBed -a <(awk -v OFS='\t' -v CHR=Chr${chr} -v GC=${lines}
'$1==CHR && $4==GC {print $1, $2, $3}' ${GC_annotation} ) -b
Output_Read_depth_Chr${chr}.bed -wb >> ReadDepth.Chr1_29.GC${lines}.dat
done < <(awk '{print $NF}' ${GC_annotation} | sort -V | uniq )

done
####
# Finally, for each "ReadDepth.Chr1_29.GC${lines}.dat" file, we
calculated mean and variance
# R Script
# Summary stats for 29 autosomes for all samples
# Calculate mean and variance per GC%
# e.g. for GC=50%
# d <- read.table("ReadDepth.Chr1_29.GC50.dat", header=T,
stringsAsFactors=F, sep='\t')
# remove columns:1-6
# dat <- d[,-c(1:6)]
# Results = t( rbind( round(apply(dat, 2, mean),2), round(apply(dat, 2,
var), 2) ))
# write.table(Results, "ReadDepth.Chr1_29.GC50.txt", row.names = FALSE,
col.names = FALSE, quote = FALSE, sep = '\t')
```

2. Phasing and Imputation

2.1 Phasing genotype likelihood (PL) data using Beagle4

```
##################################################################
# for chr in {1..29}; do
# Beagle4 (b4.r1274.jar) software is used
vcfSNPindelDEL=REF.Chr${chr}.772Animals.SNPs_DELs.PL.vcf.gz
vcfBeagle=phased.$(basename ${vcfSNPindelDEL} .PL.vcf.gz)
Beagle=b4.r1274.jar
java -Xmx10g -jar ${Beagle} \
      gl=${vcfSNPindelDEL} \
      out=${vcfBeagle} \
      burnin-its=10 \
      phase-its=15 \
      window=12000 \
      overlap=2000 \
      nthreads=6
##################################################################
```

2.2 Genotype calls using SHAPEIT2 v2.r837

```
##################################################################
GEN=$(basename ${vcfSNPindelDEL} .PL.vcf.gz).GEN
genSAM=$(basename ${vcfSNPindelDEL} .PL.vcf.gz).GEN
HAP=$(basename ${vcfSNPindelDEL} .PL.vcf.gz).HAP
hapSAM=$(basename ${vcfSNPindelDEL} .PL.vcf.gz).HAP
outMax=$(basename ${vcfSNPindelDEL} .PL.vcf.gz).ShapeITv2r837
outGraph=$(basename ${vcfSNPindelDEL} .PL.vcf.gz).ShapeITv2r837.graph
# STEP 2.2.1: Convert VCF to GEN and HAP files
# GEN/SAMPLES
bcftools convert \
      ${vcfBeagle}.vcf.gz \
      --gensample ${GEN} \
      --vcf-ids \
      --tag GP \
      --chrom Chr${chr}
# HAP/SAMPLE
bcftools convert \
      ${vcfBeagle}.vcf.gz \
      --hapsample ${HAP} \
      --vcf-ids
# Step 2.2.2 : Run SHAPEIT2 v2.r837
# --window 0.1 for WGS
# --window 2 for 50k and 777k
shapeIT2=shapeit.v2.r837/bin/shapeit
logs_ShapeIT=Chr${chr}.ShapeITv2r837.772Animals.log
${shapeIT2} \
      -call \
      --aligned \
      --input-gen ${GEN}.gen.gz ${genSAM}.samples \
      --input-init ${HAP}.hap.gz ${hapSAM}.sample \
      --effective-size 1000 \
      --input-thr 0.99 \
      --window 0.1 \
      --burn 0 \
      --run 12 \
      --prune 4 \
      --main 20 \
      --states 400 \
      --states-random 200 \
      --thread 6 \
      --output-max ${outMax}.hap.gz ${outMax}.sample \
      --output-graph ${outGraph}.gz \
      --output-log ${logs_ShapeIT}
# Step 2.2.3: Convert SHAPEIT2 haplotypes to VCF
ShapeIT_haplotypes=${outMax}.hap.gz
ShapeIT_Samples=${outMax}.sample
vcfShapeIT=${outMax}.vcf.gz
logHap2vcf=${outMax}.haps2vcf.log
${shapeIT2} \
      -convert \
      --input-haps ${ShapeIT_haplotypes} ${ShapeIT_Samples} \
      --output-vcf ${vcfShapeIT} \
      --output-log ${logHap2vcf}
##################################################################
```

2.3 Imputation using Minimac3 (Version 2.0.1)

```
##################################################################
Reference_VCF=${vcfShapeIT}
targetVCF=Chr${chr}.imputed_N_genotyped_777kAnimals.vcf.gz
outMinimac=imputed.Chr${chr}.combined777k_to_FullSeqDels
Minimac3=Minimac3/bin/Minimac3-omp
${Minimac3} \
      --refHaps ${Reference_VCF} \
      --haps ${targetVCF} \
      --prefix ${outMinimac} \
      --format GT,DS,GP \
      --myChromosome Chr${chr} \
      --log \
      --allTypedSites \
      --cpus 6
# done
##################################################################
```