



吉林大学

本科生毕业论文（设计）

中文题目：比较基因组学揭示家鸡黑白羽色变异的遗传学基础

外文题目：Comparative population genomics reveal the genetic basis underlying black and white feather color variation in domestic chickens

学生姓名 朱明远 班级 851502 学号 85150226

学 院 动物科学学院

专 业 动物科学专业

指导教师 赵志辉 职称 教授

中文摘要

随着人类生产方式由狩猎向定居转变，在几千年的过程中超过 40 余种物种被驯化，它们在驯化过程中获得了较祖先更多的表型多样性。这其中，鸡作为鸟类中多样性最多的代表，在羽毛颜色、个体大小、视觉等方面有较大的分化。传统遗传学研究多采用 QTL 定位，工作繁琐且假阳性率高，定位的区域往往很大，无法精确定位。但随着新的物理、化学实验方法的发现以及新的算法的研究取得了突破性进展，测序成本的急剧缩减，二代测序技术在此期间得到了飞速的普及与推广，因此群体基因组学这一从整个群体基因组进行遗传推断的方法得到了迅速发展。在群体基因组学的理论基础上，群体遗传学基于测序技术以及基因组图谱，应用统计学和数学原理在群体规模上对全基因组（特别是全基因组的基因型频率以及基因频率）进行研究，研究单个及复合特异性位点的效应和与遗传特异性结构相关联的全基因组效应（物种迁移、选择和必然的遗传漂变等），探究进化过程中的选择和突变作用。在重测序的基础上，鸡的许多多样性已经被解释，例如藏鸡的高原机制以及体型的大小多样性，清楚解释了鸡驯化过程中的遗传学基础。

元宝鸡是一种著名的观赏名鸡，它有着金元宝外观，从唐代以来便进行培育。元宝鸡是优良生物模型，尤其是极小体型和纯色羽毛等表型被广泛应用于研究之中。为了更好地理解影响家畜羽毛颜色变化的突变，我们采用基于 NGS 数据的比较群体基因组学研究了元宝鸡的基因组。我们将确定一系列可能影响羽色的基因突变以及区域，为在家鸡中的色素沉积模式提供新的生物学见解。

关键词：家鸡；毛色；群体基因组学；比较基因组学

Abstract

As human production patterns have changed from hunting to settlement, more than 40 species have been domesticated over the course of thousands of years, and they have gained more phenotypic diversity than their ancestors during domestication. Among them, chicken, as the most diverse representative of birds, has a large differentiation in feather color, individual size, and vision. Traditional genetic studies often use QTL mapping, which is cumbersome and has a high false positive rate. The location of the location is often large and cannot be accurately located. However, with the maturity and popularity of second-generation sequencing technology, the cost of sequencing has been reduced by nearly a hundred times. Therefore, population genomics, a method of genetic inference from the entire population genome, has developed rapidly. Based on the theory of population genomics, population genetics is based on sequencing technology and genomic maps, using statistical and mathematical principles to study whole genomes on a population scale, exploring site-specific effects and migration of gene frequencies and genotype frequencies. And the genome-wide effects of genetic drift and other genetic structures, explore the selection effects and mutations behind it. On the basis of resequencing, many of the diversity of chickens have been explained, such as the plateau mechanism of Tibetan chickens and the size and size of the body, clearly explaining the genetic basis of chicken domestication.

Yuanbao Chicken is a famous ornamental chicken. It has the appearance of Jin Yuanbao and has been cultivated since the Tang Dynasty. Yuanbao chicken is an excellent biological model, especially phenotypes such as very small size and solid color feathers are widely used in research. To better understand the mutations that affect the color change of livestock feathers, we used the comparative population genomics based on NGS data to study the genome of Yuanbaoji. We will identify a range of genetic mutations and regions that may affect feather color, providing new

insights into pigmentation patterns in chickens.

Key words: Domestic chicken; coat color; population genomics; comparative genomics

目录

第一章 文献综述.....	1
第 1 节 家鸡的驯化历史与羽毛颜色多样性的关系.....	1
第 2 节 羽毛多样性及其机制.....	1
2.1 羽毛分布与形态的多样性.....	1
2.2 羽毛颜色多样性机制.....	2
2.3 羽色多样性的遗传学定位.....	3
第二章 元宝鸡 NGS 数据的预处理.....	5
第 1 节 实验仪器及软件.....	5
1.1 主要仪器.....	5
1.2 软件以及服务器环境配置.....	5
第 2 节 实验方法.....	6
2.1 高性能服务器生物信息学开发环境.....	6
2.2 下载软件及数据库.....	6
2.3 基因组测序数据的获取.....	6
2.4 基因组比对、变异位点获取以及注释.....	7
2.5 元宝鸡的全基因组选择性清除分析策略选取.....	13
2.5.1 FST 信号选择.....	13
2.5.2 LSBL 信号选择.....	14
2.5.3 Pi 信号选择.....	21
2.5.4 SweeD 信号选择.....	23
第 3 节 实验结果.....	24
3.1 FastQC 对 clean reads 测序数据进行质量评估.....	24
3.2 ANNOVAR 注释后结果.....	25
3.3 基因密度.....	26
3.4 核苷酸多态性.....	26
3.4 LSBL 分析.....	27
3.4 核苷酸多态性.....	29
3.4 SweeD 分析.....	29
第 4 节 讨 论.....	30
第三章 正选择基因的注释及验证.....	32
第 1 节 实验软件.....	32
第 2 节 实验方法.....	32
2.1 正选择基因的注释.....	32

2.2 生物学验证.....	32
第3节 实验结果与讨论.....	33
3.1 选择性清除分析结果聚类及突变注释.....	33
3.2 chr1: 187.30Mb-188.50M 内的 Tyr 和 Rab38 可能控制黑白元宝鸡的 羽毛颜色.....	34
3.3 区域 ch1: 191.45-191.62, ch2: 69.1-72.9 Mb 作为候选区域可能 影响毛色.....	34
结 论.....	36
致谢	37
参考文献	38

第一章 文献综述

第 1 节 家鸡的驯化历史与羽毛颜色多样性的关系

鸡在生物分类学上属于脊索动物门、鸟纲、鸡形目，鸡亚目，隶属于雉科下的原鸡属(*Gallus*)，目前普遍认为驯化约起始于 6000 年，属内包括绿原鸡(*Gallus varius*)、红原鸡(*Gallus gallus*)、锡兰原鸡(*Gallus lafayettei*)、灰纹原鸡(*Gallus sonnerati*)这四种被科研界所共识的原鸡。线粒体内切酶多态性分析、混池重测序分析等表明，鸡的驯化历史比想象中的更加复杂。到目前为止，无论是形态学、解剖学、还是遗传学、分子生物学、生物化学等方面对于家鸡的起源问题(单起源和多起源)及具体确切的起源地、起源时间等都无法得到确切的答案，至今仍具有较大争议^[1]。

驯养动物是人类的生活从狩猎到定居、生活质量获得巨大改善的重要一步，目前已经有超过 40 多种物种被成功驯化。随着人类社群的发展壮大，鸡的养殖规模以及饲养区域的不断扩大，在适应不同地理与生态环境、人类的偏好选择，家鸡的在进化过程中获得了较他们的祖先乃至其他鸟类更多的表型多样性。经过了几千年的繁殖与驯化，鸡的品种在体型、羽毛类型与颜色、皮肤颜色以及冠的类型方面形成了较大差别^[2,3,4]。

人类的认知以及习俗—特别是对于羽毛颜色的特殊喜好选择促进了特殊地域市场上羽毛颜色的多样性的增加或者减少，人们在购买鸡时较为关注鸡的羽毛、胫的颜色等表型特征，这一消费的习惯很大程度上影响了市场。因此对于家禽育种研究者而言，为了适应消费者的需求和市场供应，研究重点就是探究羽毛颜色表型背后的遗传学基础^[5]。此外，研究羽毛颜色的表型与基因型的关系对于探究家鸡的驯化与起源遗传资源的保护、分子遗传学育种等方面也都具有重要的意义。

第 2 节 羽毛多样性及其机制

2.1 羽毛分布与形态的多样性

羽毛是鸟类最复杂的角化结构之一，羽毛在鸟类在各个生长发育的阶段起着至关重要的作用，例如伪装躲避天敌以及物种的繁衍。羽毛作为一个结构复杂的

组织器官，其遗传多样性为进化以及发育生物学提供了良好的生物模型^[6,7]。

家鸡在羽毛在其生长发育和分化的每一阶段，都显示了其丰富的多样性，常见的有羽毛分布不同（例如全部皮肤裸露无毛，裸颈，毛脚等）、颜色不一（如棕色，灰色，黑色，白色，绿色，粉红色，黄色或橙色等）和羽毛结构不一致（如卷毛，丝羽片羽等）。近年来关于羽毛的研究取得了巨大的进展，包括裸颈特征是由 BMP12 基因引起的，它改变了羽毛的在颈部的分布；黄色色素羽毛性状是由 MuPKS 基因引起的黄色色素的积累；而卷曲羽毛特性是由 KRT75 基因引起的，它在特征卷曲的羽毛轴和倒钩中起着重要作用。

2.2 羽毛颜色多样性机制

自然野生状态下，羽毛颜色主要起着保护色、求偶交配等作用。而家养鸡从早期的驯化开始，出于方便进行畜产品处理和纯种繁殖育种以及观赏、祭祀等用途，经过长期的人工选择以及对饲料和环境的适应，与其野生祖先相比展现出了巨大的表型多样性^[8,9]。

有关色素沉积的研究主要集中在小鼠身上，研究证明在小鼠身上近 400 个基因的突变直接或者间接影响皮肤及毛发颜色。鸟类的羽毛凭借其表型的多样性，作为研究色素沉积分布的良好模型，在该领域取得了不错的进展。

羽毛颜色的影响主要包含各种色素的比例、种类、分布、数量影响的化学颜色，以及羽毛上皮表面的物理结构所导致的结构色^[10,11]。鸡中的大量色素沉着基因已经被发现证实。对于单基因而言，部分毛色的基因已经通过传统 QTL、比较基因组学、全基因组扫描定位，特别是黑素皮质素受体 1(MC1R)、细胞周期依赖性蛋白激酶抑制剂 2A(cDKN2A)、水溶载体蛋白 45 家族第 2 成员(ALC45A2)以及后续通过连锁互换及下游试验验证的 PMEL17、酪氨酸酶基因、sox10 及 melanophilin 基因等，已被公认为研究羽色的目标基因。研究表明，虽然羽色的遗传涉及复杂的一系列单基因突变或多基因互作，但绝大多数的色素沉积都与公认的一种色素通路相关，分别是类胡萝卜素和真黑色素的产生调节通路。类胡萝卜素，作为一种色素由植物、微生物体内合成（动物不能直接合成，只能通过摄入进行转换），通过作为光的吸收色素发挥介质功能，保护光合作用相关酶以及底物不受到直接能量冲击并延长其使用寿命。虽然该物质不能在鸟类体内合成，但可通过摄入相关食物在体内进行堆积、转化、分布，参与蛋黄、虾蟹的几丁质

外壳、羽毛等颜色组成。国内外研究表明, **BCMO1** 以及 **BCO2** (β -胡萝卜素加氧酶 2) 的突变显著改变了类胡萝卜素的分布^[12,13,14], 进而影响黄色羽毛、皮肤和喙的表型。

基于黑色素的色素沉积研究较为深入, 在哺乳动物中研究与褐黑素色素有关的基因(例如 **ASIP**, **MC1R**, **Tyr**, **Tyrp1**, **Tyrp2**) 的信息, 同样在鸟类中适用。羽毛中的黑色素与哺乳动物类似, 是一种真黑色与褐黑素的混合物, 混合的比例决定了羽毛颜色从纯黑到纯白的不同变化。

黑色素的调控网络机制极其复杂, 涉及多个通路的联动。它的生物合成起始于酪氨酸酶(**TYR**)对酪氨酸羟化催化, 之后参与不同的调节网络, 与众多基因的有序表达息息相关^[15]。

MC1R 蛋白是一种广泛存在的 G 蛋白偶联受体, 在细胞中作为传导物质, 通过接受 **MSH**(黑素细胞刺激素)的上游信号进行激活, 引起 **AC**(腺苷酸环化酶)活化, 介导下游的 **cAMP** 信号通路的信号传导, 激活酪氨酸酶蛋白家族相关蛋白(例如 **TYRP1**), 促使例如褐黑素小体等合成真黑素^[16]。**Agouti** 信号蛋白由 **ASIP** 基因编码, 被认为在下游通路中发挥 **MC1R** 的反向激动剂作用, 研究表明它的表达量与众多的动物体内毛色过黑或白化表型相关, 原理可能作为 α -**MSH** 及 **ACTH** 竞争性抑制剂, 竞争性地结合 **MC1R** 受体并不产生任何的生物学效应以及酶构象的改变, 通过减少 **MA1R** 受体的响应数量进而引起 **cAMP** 表达水平的下调, 并通过一定的负反馈机制, 有调控的减少下游通路真黑素的产生。

2.3 羽色多样性的遗传学定位

前人通过 QTL 定位、全基因组扫描等方法定位到了许多与羽色相关的基因, 例如酪氨酸酶(**TYR**)、酪氨酸酶相关蛋白 1(**TYRP1**)以及通过群体遗传学及下游试验定位到的黑素皮质激素受体 1(**MC1R**)等基因, 因此下文将着重对这些基因近年来的相关遗传学进展、研究的过程及方法进行综述。

酪氨酸酶(**TYR**)是黑色素合成途径催化反应速度最慢的关键酶, 起着限速酶的作用, 酶活性与黑色素的合成的下游通路息息相关, 通过影响活性高低从而改变毛色的深浅。在许多家鸡中, 例如坝上长尾鸡, 相关研究表明部分启动子的活性会由于 **TYR** 基因的突变受到影响, 从而导致转录以及下游的黑色素合成受到干扰, 导致白化以及其他的羽毛白色变异表型的出现。同时, **TYRP1** 基因作

为较早进行研究的色素基因，通过在黑色素下游合成途径中作为二羟基吲哚酸酶，发挥其酶活性进而与 TYR 基因的表达密切相关^[17,18]。

研究人员通过 BAC 文库以及微卫星标记定位到了基因，通过表达水平以及突变个体对比发现 TYRP1 基因作为黑色素合成的关键酶之一对丝羽乌骨鸡和白来航鸡毛色、表皮具有显著影响^[19]。

黑皮质素 1 受体 (MC1R) 基因是黑色素合成的重要调控基因，它经常被用来作为候选基因在重要的经济动物鸡的重要遗传育种的意义。家鸡中的黑素皮质素受体基因被首次分离并克隆于 19 世纪，并发现与哺乳动物 MC1R 基因具有较高的同源性^[21]，之后通过对比 MC1R 基因型与羽色表型的对应关系，类比了哺乳动物的调控机制与小鼠中的毛色研究，猜测 MC1R 变异对应着不同的家鸡羽色^[20]。

此外，MC1R 基因的变异很可能与趋同进化过程中动物羽色的选择有关。在前人的研究中，研究者通过设计引物提取组织进行 sanger 测序，比较在云南某地 200 只天鹅 MC1R 基因的所有基因突变，并且记录羽色，通过两者之间关系构建了系统发育树，发现进化的分支可能存在不止一个的复杂过程。同样，在猫科动物中也得到了相似的结果^[21]。

许多突变位点与表型的对应关系也逐渐被探究出来，通过对比红原鸡与白来航商品鸡，结果发现 MC1R 的突变-E92K 和 M71T 对应着鸡的全黑色羽色。

目前，在全基因组水平上对家鸡品种内群体遗传变异-例如进行选择清除分析是探究羽色遗传学基础的重要一环，这对畜禽的种质特性评估起到显著的推动作用，在新的生物技术以及全基因组更大水平上进行探究加快了研究的进展，是优质禽类遗传种质资源科学利用、保存的有效方案^[22]。

虽然黑色素，卟啉，类胡萝卜素以及多烯等已经被证实与毛色相关，但具体的突变基因以及相关机理尚未被证实。前人的研究材料主要集中于含有少量变异的商品纯和鸡，他们的遗传物质较为纯和、含有较少的突变，并不能完整观察到控制毛色突变的大量的因素，因此给研究毛色相关的生物学家以及遗传育种工作者造成了很多困扰。

第二章 元宝鸡 NGS 数据的预处理

第 1 节 实验仪器及软件

1.1 主要仪器

高性能计算平台（high performance computer）：中国科学院昆明动物所研究所；

阿里云服务器；遗传育种实验室塔式服务器；

1.2 软件以及服务器环境配置

高性能计算平台（high performance computer）：系统使用 Torque2.4.8 和 maui3.3 作为机群作业调度管理软件，可以支持 288 个计算核的并行计算作业，节点可以处理最大 512GB 内存需求的数据分析

表 1 超算已经安装的公用基础软件和部分应用软件清单：

部分公用基础软件
MPICH, Openmpi2, Perl 5.12（包含各种模块），Python 2.6.5（包含各种模块）， fftw3.1.2, atlas, lapack,blas, INTEL-COMPLIER, PGI compiler, matlab, confuse, cairo,apache, mysql, imagemagick,pango, torque,maui, fontconfig, freetype,rrdtool, sar2rrd, pango, pixman, pkgconfig, pigz, mpi4py, Parallel::MPI(mpi implementation on perl), libpng, libxml, glasgow haskell compiler, f95,gfortran, zlib, java, R(包含各种模块),
部分应用软件
Phyml, RAxML-7.2.5, Mrbayes, Parallel_MrBayes, Qiime, GAPipeline, Illumina_Genome_Analyzer, CASAVA, Mireap_0.2, Muscle, Blast, Mpiblast, Abyss, codeml, overlapp, dnadist, dnastar, namd, charm++, blat, mpiblast, tigcl, muscle, DBD-SQLite-1.31, DBD-mysql-4.018, Data-ShowTable-3.3, DBI-1.616, Math-CDF-0.1, prank, paml44, boost1_46_1, cufflinks1.0.3, clustalw, tophat, myrna-1.0.4, amos-3.0.0, bowtie, hapsembler, SOAPdenovo-V1.05, samtools, velvet_1.1.04, Phrap, GapCloser, idba, bwa-0.5.9, MTR, pysam-0.4.1, biopython, bioperl, cd-hit-v4.3, dna_blast, inparanoid, mrbayes, protest, mauve, hmmer, phylip, lastz, Mummer3, Soap2, Velvet, Cap3, T-COFFEE_distribution_Version_8.99

第 2 节 实验方法

2.1 高性能服务器生物信息学开发环境

硬件环境：Dell PowerEdge R43, 双 CPU 服务器, 内存:16G;硬盘:300G;15K 阵列卡, 电源:495W;

操作系统：为 CentOS 7 (<https://www.centos.org>) ;

2.2 下载软件及数据库

(1) FastQC 软件基于 Java 编写, 它可以通过一键式可视化的方式快速地对 fastq、fasta 测序数据进行质量评估, 从

<http://www.bioinformatics.babraham.ac.uk/projects/fastqc/> 获取

(2) 其余软件代码均收录入 GitHub, 从 <https://github.com/csf-ngs/fastqc> 获取

2.3 基因组测序数据的获取

分析数据所有 24 只鸡基因组均来自文章 “Comparative population genomics reveals genetic basis underlying body size of domestic chickens” 中的部分数据集

(<https://academic.oup.com/jmcb/article/8/6/542/2527549>) , 该研究由 8 只黑色元宝鸡, 8 只白色元宝鸡以及 8 只其他家养鸡组成。通过文末提供的 SRA 号, SRAtoolkit 工具批量下载 NCBI 的数据集, 存储于高性能服务器。

表 2 实验所用的元宝鸡的测序数据

家鸡的毛色及类型	样本编号	SRA 下载号
yuanbao_black	140401_Ypt581-700_AGTCAG_L005	SRP034930
yuanbao_black	140401_Ypt582-700_CCGTCC_L005	SRP034930
yuanbao_black	140403_Ypt588-650_GCCAAT_L006	SRP034930
yuanbao_black	140430_Ypt589-650_CAGATC_L004	SRP034930
yuanbao_black	140403_Ypt595-700_GCCAAT_L003	SRP034930
yuanbao_black	140403_Ypt596-700_CAGATC_L004	SRP034930
yuanbao_black	140403_Ypt597-700_ACTTGA_L004	SRP034930
yuanbao_black	140403_Ypt598-700_GATCAG_L004	SRP034930
yuanbao_white	140401_Ypt578-700_GCCAAT_L004	SRP034930
yuanbao_white	140401_Ypt580-700_CTTGTA_L005	SRP034930
yuanbao_white	140403_Ypt583-650_ATCACG_L005	SRP034930
yuanbao_white	140403_Ypt584-650_CGATGT_L005	SRP034930
yuanbao_white	140403_Ypt591-700_CGATGT_L003	SRP034930
yuanbao_white	140403_Ypt593-700_TGACCA_L003	SRP034930
yuanbao_white	140403_Ypt594-700_ACAGTG_L003	SRP034930

yuanbao_white	140430_YPt601-700_CTTGTA_L001	SRP034930
yuanbao_other	140403_Ypt585-650_TTAGGC_L005	SRP034930
yuanbao_other	140403_Ypt586-650_TGACCA_L005	SRP034930
yuanbao_other	140416_Ypt590-650_ATCACG_L004	SRP034930
yuanbao_other	140401_Ypt579-700_CAGATC_L004	SRP034930
yuanbao_other	140403_Ypt587-650_ACAGTG_L005	SRP034930
yuanbao_other	140403_Ypt592-700_TTAGGC_L003	SRP034930
yuanbao_other	140403_Ypt599-700_TAGCTT_L004	SRP034930
yuanbao_other	140403_YPt600-700_GGCTAC_L004	SRP034930

2.4 基因组比对、变异位点获取以及注释

(1) 24 只家鸡的原始的测序首先使用 Cutadapt 以及 FastQC 进行去除接头及数据的质量控制,采用 Btrim 去掉不合乎质量要求的碱基以及序列。参考 BWA 的说明文档,使用索引命令 `bwa index fai`, 构建家鸡最新参考基因组 (Galgal4) 的索引,为后续的比对分析做准备。参考试验数据的长度以及文件大小后,我们选取 BWA 三个算法 (BWA backtrack, BWA SW 和 BWA MEM) 中适合短序列大数据的 MEM 比对算法,该算法将标准化处理后的双端测序读长 (reads) 比对到之前构建的参考基因组 index 上,除了算法、bam 压缩命令以及数目设置外,其他的参数均按照模板案例设置为默认参数^[23]。

对于生成的 bam 文件,我们采用 GATK 软件包 (Genome Analysis Toolkit) 以及其中内置的 Picards 对符合 SAM 及 VCF 文件规范的数据进行格式处理及转置。在进行读长排序、去除数据中的重复读长后,重新比对 INDEL 附近的读长并根据数据的实际情况矫正筛选参数,通过定制的参数修改方案来控制对比质量。接下来使用 GATK 的 UnifiedGenotyper 用于 Variant calling,在并行算法中设定 variant 位点的置信阈值,若变异低于该阈值则标记为 low quality 并不计入报告,本次试验中考虑样本量采用贝叶斯最大似然模型进行处理,通过计算获得群体 SNP、基因型以及基因频率,通过后验概率对单个家鸡样本的单点突变和基因型进行评估。生成的 vcf 文件我们使用 VariantFiltration 加入更多注释信息,在对 SNP 进行过滤后使用 ANNOVAR 软件注释后续文件,生成 vcf 注释文件。

我们将以上步骤所有流程进行合并,通过管道进行传输,脚本如下。注释后的 vcf 文件,我们导出其染色体号、起始终止位置,以饼图的形式展示变异类型。

```

1. @
2. FULL SCRIPT
3.
4. #!/bin/sh
5. #PBS -N s4328_HJCMCXX_L2_map
6. #PBS -q small
7. #PBS -l nodes=1:ppn=4
8. #PBS -o /lustre/user/newton/qcjob.out
9. #PBS -e /lustre/user/newton/qcjob.err
10.
11. cd /lustre/user/newton/africanchicken
12.
13. #for i in s4328_HJCMCXX_L2
14. #do
15. gzip -d s4328_HJCMCXX_L2_1.clean.fq.gz
16. gzip -d s4328_HJCMCXX_L2_2.clean.fq.gz
17.
18. ##Quality control [Trim paired ends]
19.
20. /lustre/user/shengk/shengk_bin/software/Btrim64 -S -q -s summary_s4328_HJCMCXX_L2_1.clean.fq.Btrim64 -l 25 -a 20 -o s4328_HJCMCXX_L2_1.clean.fq.Btrim64 -t s4328_HJCMCXX_L2_1.clean.fq
21. /lustre/user/shengk/shengk_bin/software/Btrim64 -S -q -s summary_s4328_HJCMCXX_L2_2.clean.fq.Btrim64 -l 25 -a 20 -o s4328_HJCMCXX_L2_2.clean.fq.Btrim64 -t s4328_HJCMCXX_L2_2.clean.fq
22. gzip s4328_HJCMCXX_L2_2.clean.fq s4328_HJCMCXX_L2_1.clean.fq
23. perl /lustre/user/shengk/shengk_bin/paired_end_trim.pl summary_s4328_HJCMCXX_L2_1.clean.fq.Btrim64 summary_s4328_HJCMCXX_L2_2.clean.fq.Btrim64 s4328_HJCMCXX_L2_1.clean.fq.Btrim64 s4328_HJCMCXX_L2_2.clean.fq.Btrim64
24.
25. rm summary_s4328_HJCMCXX_L2_1.clean.fq.Btrim64 summary_s4328_HJCMCXX_L2_2.clean.fq.Btrim64 s4328_HJCMCXX_L2_1.clean.fq.Btrim64 s4328_HJCMCXX_L2_2.clean.fq.Btrim64
26.
27.
28. ++++++
29. + to prepare the reference genome +
30. ++++++
31.
32. #for "Gallus_gallus.Gallus_gallus-5.0.dna.toplevel.fa.gz" [index and create dictionary]

```

```

33.
34. cd /lustre/user/newton/Galgal5
35. gzip -d Gallus_gallus.Gallus_gallus-5.0.dna.toplevel.fa.gz
36.
37. /lustre/user/newton/software/samtools-1.3/samtools faidx Gallus_gallus.Gallu
   s_gallus-5.0.dna.toplevel.fa
38. /lustre/user/newton/software/bwa-0.7.12/bwa index Gallus_gallus.Gallus_gallu
   s-5.0.dna.toplevel.fa
39. /lustre/user/newton/software/jre1.8.0_60/jre1.8.0_60/bin/java -Xmx2g -jar /l
   lustre/user/newton/software/picard-tools-1.119/CreateSequenceDictionary.jar R
   =./Gallus_gallus.Gallus_gallus-5.0.dna.toplevel.fa OUTPUT=Gallus_gallus.Gall
   us_gallus-5.0.dna.toplevel.dict
40.
41. ##mapping pe
42.
43. /lustre/user/newton/software/bwa-0.7.12/bwa mem -t 12 -M -R "@RG\tID:s4328_
   HJCMLCCXX_L2\tLB:s4328_HJCMLCCXX_L2\tPL:ILLUMINA\tSM:s4328_HJCMLCCXX_L2" /lu
   stre/user/newton/Galgal5/Gallus_gallus.Gallus_gallus-5.0.dna.toplevel.fa s43
   28_HJCMLCCXX_L2_1.clean.fq.Btrim64.pe s4328_HJCMLCCXX_L2_2.clean.fq.Btrim64.
   pe | /lustre/user/newton/software/samtools-1.3/samtools view -bS - -o s4328_
   HJCMLCCXX_L2_pe.bam
44.
45. cat s4328_HJCMLCCXX_L2_1.clean.fq.Btrim64.se s4328_HJCMLCCXX_L2_2.clean.fq.B
   trim64.se > s4328_HJCMLCCXX_L2.fq.se
46. gzip s4328_HJCMLCCXX_L2_1.clean.fq.Btrim64.pe s4328_HJCMLCCXX_L2_2.clean.fq.
   Btrim64.pe s4328_HJCMLCCXX_L2_1.clean.fq.Btrim64.se s4328_HJCMLCCXX_L2_2.cle
   an.fq.Btrim64.se
47.
48. ##mapping se
49.
50. /lustre/user/newton/software/bwa-0.7.12/bwa mem -t 12 -M -R "@RG\tID:s4328_
   HJCMLCCXX_L2_se\tLB:s4328_HJCMLCCXX_L2\tPL:ILLUMINA\tSM:s4328_HJCMLCCXX_L2"
   /lustre/user/newton/Galgal5/Gallus_gallus.Gallus_gallus-5.0.dna.toplevel.fa
   s4328_HJCMLCCXX_L2.fq.se | /lustre/user/newton/software/samtools-1.3/samtool
   s view -bS - -o s4328_HJCMLCCXX_L2_se.bam
51.
52. ##index pe & se bam files [this command seems to have a problem]
53. #/lustre/user/newton/software/samtools-1.3/samtools index s4328_HJCMLCCXX_L2
   _pe.bam
54. #/lustre/user/newton/software/samtools-1.3/samtools index s4328_HJCMLCCXX_L2
   _se.bam
55.
56. ##trying to sort first?? Sorting bam file [have to put options okay!]

```

```

57. *** /lustre/user/newton/software/samtools-1.3/samtools sort /lustre/user/new
    ton/4275_clean_aln.bam -o 4275_clean_aln.sorted
58. *** /lustre/user/newton/software/samtools-1.3/samtools sort /lustre/user/new
    ton/4275_clean_aln.bam -o 4275_clean_aln.sorted
59.
60. ###pe #### [sort & mark duplicates & index]
61. /lustre/user/newton/software/jre1.8.0_60/jre1.8.0_60/bin/java -Xmx4g -jar /l
    lustre/user/newton/software/picard-tools-1.119/SortSam.jar INPUT=s4328_HJCLC
    CXX_L2_pe.bam OUTPUT=s4328_HJCLCXX_L2_pe.sort.bam SORT_ORDER=coordinate
62. /lustre/user/newton/software/jre1.8.0_60/jre1.8.0_60/bin/java -Xmx4g -jar /l
    lustre/user/newton/software/picard-tools-1.119/MarkDuplicates.jar INPUT=s432
    8_HJCLCXX_L2_pe.sort.bam OUTPUT=s4328_HJCLCXX_L2_pe.sort.dedup.bam METRI
    CS_FILE=s4328_HJCLCXX_L2_pe.dedup.metrics VALIDATION_STRINGENCY=LENIENT MA
    X_FILE_HANDLES_FOR_READ_ENDS_MAP=50
63. /lustre/user/newton/software/samtools-1.3/samtools index s4328_HJCLCXX_L2_
    pe.sort.dedup.bam
64.
65. ##REQUIRE KNOWN FILE...
66. #original file has diff. chr. order compared to the galgal5 ref.
67. #to change the order for compatibility use:
68.
69. #!/bin/sh
70. #PBS -N change.order.quick
71. #PBS -q small
72. #PBS -l nodes=1:ppn=1
73. #PBS -o /lustre/user/newton/Galgal5/choder.out
74. #PBS -e /lustre/user/newton/Galgal5/choder.err
75.
76. cd /lustre/user/newton/Galgal5
77.
78. grep -P "^#" Gallus_gallus.vcf > neworder001.vcf
79.
80. for orders in 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
    25 26 27 28 30 31 32 33 W Z MT LGE64; do
81. grep -P "^${orders}\$" Gallus_gallus.vcf >> neworder001.vcf
82. done
83.
84.
85. *****
86. ##pe [realigh target & realign indels & recalibrate base & print reads]
87.
88. /lustre/user/newton/software/jre1.8.0_60/jre1.8.0_60/bin/java -Xmx16g -jar /
    lustre/user/newton/software/GenomeAnalysisTK34.jar -R /lustre/user/newton/Ga
    lgal5/Gallus_gallus.Gallus_gallus-5.0.dna.toplevel.fa -T RealignerTargetCrea

```



```

tor -o s4328_HJCMLCCXX_L2_pe.sort.dedup.bam.intervals -I ./s4328_HJCMLCCXX_L
2_pe.sort.dedup.bam -known /lustre/user/newton/Galgal5/neworder001.vcf
89.
90. /lustre/user/newton/software/jre1.8.0_60/jre1.8.0_60/bin/java -Xmx16g -jar /
lustre/user/newton/software/GenomeAnalysisTK34.jar -R /lustre/user/newton/Ga
lgal5/Gallus_gallus.Gallus_gallus-5.0.dna.toplevel.fa -T IndelRealigner -tar
getIntervals ./s4328_HJCMLCCXX_L2_pe.sort.dedup.bam.intervals -o ./s4328_HJC
MLCCXX_L2_pe.sort.dedup.realign.bam -I ./s4328_HJCMLCCXX_L2_pe.sort.dedup.ba
m -known /lustre/user/newton/Galgal5/neworder001.vcf
91.
92. /lustre/user/newton/software/jre1.8.0_60/jre1.8.0_60/bin/java -Xmx16g -jar /
lustre/user/newton/software/GenomeAnalysisTK34.jar -R /lustre/user/newton/Ga
lgal5/Gallus_gallus.Gallus_gallus-5.0.dna.toplevel.fa -T BaseRecalibrator -o
./s4328_HJCMLCCXX_L2_pe.sort.dedup.realign.bam.recal.grp -I ./s4328_HJCMLCC
XX_L2_pe.sort.dedup.realign.bam -knownSites /lustre/user/newton/Galgal5/newo
rder001.vcf
93.
94. /lustre/user/newton/software/jre1.8.0_60/jre1.8.0_60/bin/java -Xmx16g -jar /
lustre/user/newton/software/GenomeAnalysisTK34.jar -R /lustre/user/newton/Ga
lgal5/Gallus_gallus.Gallus_gallus-5.0.dna.toplevel.fa -T PrintReads -I ./s43
28_HJCMLCCXX_L2_pe.sort.dedup.realign.bam -BQSR ./s4328_HJCMLCCXX_L2_pe.sort
.dedup.realign.bam.recal.grp -o ./s4328_HJCMLCCXX_L2_pe.sort.dedup.realign.r
ecal.bam
95.
96. ###se ####
97. #[sort & mark duplicates & index]
98.
99. /lustre/user/newton/software/jre1.8.0_60/jre1.8.0_60/bin/java -Xmx4g -jar /l
ustre/user/newton/software/picard-tools-1.119/SortSam.jar INPUT=s4328_HJCMLC
CXX_L2_se.bam OUTPUT=s4328_HJCMLCCXX_L2_se.sort.bam SORT_ORDER=coordinate
100. /lustre/user/newton/software/jre1.8.0_60/jre1.8.0_60/bin/java -Xmx4g -jar /
lustre/user/newton/software/picard-tools-1.119/MarkDuplicates.jar INPUT=s43
28_HJCMLCCXX_L2_se.sort.bam OUTPUT=s4328_HJCMLCCXX_L2_se.sort.dedup.bam METR
ICS_FILE=s4328_HJCMLCCXX_L2_se.dedup.metrics VALIDATION_STRINGENCY=LENIENT M
AX_FILE_HANDLES_FOR_READ_ENDS_MAP=50
101. /lustre/user/newton/software/samtools-1.3/samtools index s4328_HJCMLCCXX_L2
_se.sort.dedup.bam
102.
103. #[realign target & realign indels & recalibrate base & print reads]
104.
105. /lustre/user/newton/software/jre1.8.0_60/jre1.8.0_60/bin/java -Xmx16g -jar
/lustre/user/newton/software/GenomeAnalysisTK34.jar -R /lustre/user/newton/G
algal5/Gallus_gallus.Gallus_gallus-5.0.dna.toplevel.fa -T RealignerTargetCre

```

```

ator -o s4328_HJCMLCCXX_L2_se.sort.dedup.bam.intervals -I ./s4328_HJCMLCCXX_
L2_se.sort.dedup.bam -known /lustre/user/newton/Galgal5/neworder001.vcf
106. /lustre/user/newton/software/jre1.8.0_60/jre1.8.0_60/bin/java -Xmx16g -jar
/lustre/user/newton/software/GenomeAnalysisTK34.jar -R /lustre/user/newton/
Galgal5/Gallus_gallus.Gallus_gallus-5.0.dna.toplevel.fa -T IndelRealigner -
targetIntervals ./s4328_HJCMLCCXX_L2_se.sort.dedup.bam.intervals -o ./s4328_
HJCMLCCXX_L2_se.sort.dedup.realigh.bam -I ./s4328_HJCMLCCXX_L2_se.sort.dedup
.bam -known /lustre/user/newton/Galgal5/neworder001.vcf
107. /lustre/user/newton/software/jre1.8.0_60/jre1.8.0_60/bin/java -Xmx16g -jar
/lustre/user/newton/software/GenomeAnalysisTK34.jar -R /lustre/user/newton/
Galgal5/Gallus_gallus.Gallus_gallus-5.0.dna.toplevel.fa -T BaseRecalibrator
-o ./s4328_HJCMLCCXX_L2_se.sort.dedup.realigh.bam.recal.grp -I ./s4328_HJC
MLCCXX_L2_se.sort.dedup.realigh.bam -knownSites /lustre/user/newton/Galgal5/
neworder001.vcf
108. /lustre/user/newton/software/jre1.8.0_60/jre1.8.0_60/bin/java -Xmx16g -jar
/lustre/user/newton/software/GenomeAnalysisTK34.jar -R /lustre/user/newton/G
algal5/Gallus_gallus.Gallus_gallus-5.0.dna.toplevel.fa -T PrintReads -I ./s
4328_HJCMLCCXX_L2_se.sort.dedup.realigh.bam -BQSR ./s4328_HJCMLCCXX_L2_se.so
rt.dedup.realigh.bam.recal.grp -o ./s4328_HJCMLCCXX_L2_se.sort.dedup.realigh
.recal.bam
109.
110. #####merge
111. #[merge pe & se, index, calculate depth]
112.
113. /lustre/user/newton/software/jre1.8.0_60/jre1.8.0_60/bin/java -Xmx16g -jar
/lustre/user/newton/software/picard-tools-1.119/MergeSamFiles.jar INPUT=./s
4328_HJCMLCCXX_L2_pe.sort.dedup.realigh.recal.bam INPUT=./s4328_HJCMLCCXX_L2
_se.sort.dedup.realigh.recal.bam OUTPUT=s4328_HJCMLCCXX_L2.sort.rmdup.realn
.recal.merge.bam
114.
115. /lustre/user/newton/software/samtools-1.3/samtools index s4328_HJCMLCCXX_L2
.sort.rmdup.realn.recal.merge.bam
116. /lustre/user/newton/software/samtools-1.3/samtools depth s4328_HJCMLCCXX_L2
.sort.rmdup.realn.recal.merge.bam | awk '{sum+=$3} END { print "19912.bam A
verage depth = ",sum/NR}' >./s4328_HJCMLCCXX_L2.all.depth.txt
117. /lustre/user/newton/software/samtools-1.3/samtools depth s4328_HJCMLCCXX_L2
.sort.rmdup.realn.recal.merge.bam -r Z | awk '{sum+=$3} END { print "19912_
chrZ.bam Average depth = ",sum/NR}' >./s4328_HJCMLCCXX_L2.chrZ.depth.txt
118.
119. #done

```

最后我们选择 kb 作为单位，对 24 个个体在每个染色体的基因密度进行绘图，代码如下

```
1. library(ggplot2)
```

```
2. rm(list=ls())
3. setwd("C:\\Users\\Administrator\\Desktop\\figure1")
4. getwd()
5. mydata=read.table("yuanbao_24total..recode.vcf.windowed.pi", header=T)
6. mydata$CHROM <- factor(mydata$CHROM, levels=unique(mydata$CHROM))
7. mydata=mydata[order(as.numeric(mydata[,1]),decreasing = FALSE),]
8. ggplot(mydata,aes(CHROM,PI,fill=CHROM,alpha=1/30))+
9.   geom_boxplot(position="identity")
```

2.5 元宝鸡的全基因组选择性清除分析策略选取

我们为了充分挖掘在家鸡驯化过程中潜在的人工选择信号，找寻相关基因，我们比较了常用的选择性清除分析策略后，采用基于不同算法的 F_{ST} 、Pi、LSBL、SweeD 四种方法对黑白元宝鸡群体基因组数据进行比较，通过其他的家鸡基因组数据作为参照过滤掉无关的信号。

2.5.1 F_{ST} 信号选择

群体固定指数(F_{ST})隶属于 F 统计量，常用来衡量总群体 (T) 与亚群体 (S) 之间在遗传学上的差异问题，取值范围在 0~1 之间，数值越小代表差异越小。这里我们采取的是 F_{ST} 的无偏估计量，该方法对不同亚群体进行分析，利用测序所统计出来的各个亚群体基因频率数据和各个亚群体的大小计算出 F_{ST} 估计量^[24]。具体步骤如下

(1)24 只元宝鸡的 vcf 文件传输至高性能服务器，24 只元宝鸡文件根据上一步合并为了一个 vcf 大文件

(2)根据个体信息分别将元宝黑羽鸡、元宝白羽鸡、其他家鸡根据个体名称导出至单独文件夹，命名为 A.list,B.list,C.list

(3)使用 vcftools 工具的群体间 fst 功能，以三个群体个体名称为索引，分别计算 vcf 中三个群体之间位点的 fst 指数，命名为 A-B.site.fst、B-C.site.fst、A-C.site.fst

```
1. biosoft/vcftools/vcftools_0.1.13/bin/vcftools --gzvcf Sailama.chr2.recode.vcf.gz --weir-fst-pop A.list --weir-fst-pop B.list --out A-B.site.fst
```

生成结果应该符合以下格式

1.	CHROM	POS	WEIR_AND_COCKERHAM_FST
2.	1	545	-nan
3.	1	734	0.0569106
4.	1	765	0.0569106
5.	1	773	0.0569106
6.	1	1587	-nan
7.	1	2965	0.153478
8.	1	2973	0.226479
9.	1	5694	-0.14
10.	1	5726	-nan

2.5.2 LSBL 信号选择

LSBL (locus-specific branch length)也是一种基于种群分化的方法，如果某个基因在一个种群中受到选择，而在另一个当中没有受到选择，该基因就会在两个种群之间呈现出很大的频率差异，这种差异会比未受到选择的基因表现出的频率差异更加显著。

(1) 使用 vcftools 根据亚群间的名称列表，拆分 vcf 文件为三个 vcf

```
bioinfo/vcftools/vcftools_0.1.13/bin/vcftools --gzvcf Sailama.chr2.recode.vcf.gz
--keep A.list --out salama. --recode & ##a 的样本个体在 a.list,取出 a
```

(2) 使用 perl 语言脚本，对两两之间的 fst 值进行过滤，去除空集

```
open f1,"<B-C.site.fst.weir.fst";

open f2,">B-C.site.fst.weir.fst..Nonan.site.fst";

while(<f1>){    chomp;

    if ($_ =~ /nan|POS/){

        ;    }

    else{

        print f2 "$_\n";

    }    }
```

(3)使用 perl 脚本，将三个文件的交集取出来，并单独存储命名

```
1. open A,"<a.txt";
2. foreach (<A>) { @pairs=split(/ /, $_);
```

```

3.  $count=@pairs;
4.  $linea=$pairs[1];
5.  $hasha{$linea} = $pairs[2];
6.  }
7.  close(A);
8.  open B,"<b.txt";
9.  open m2,">m2.txt";
10. foreach (<B>) { @pairs=split(/ /, $_);
11.  $count=@pairs;
12.  $lineb=$pairs[1];
13.  $hashb{$lineb} = $pairs[2];
14. }
15. close(B);
16. open C,"<c.txt";
17. foreach (<C>) { @pairs=split(/ /, $_);
18.  $count=@pairs;
19.  $linec=$pairs[1];
20.  $hashc{$linec} = $pairs[2];
21. }
22. close(C);
23. ###把 ABC 三个文件扔给 hash 数组
24. ##
25. open(B,"<b.txt");
26. open m1,">m1.txt";
27. while(<B>){
28. @pairs=split(/ /, $_);
29.  $line4=$pairs[2];
30.  if(exists $hasha{$line4}){
31.    print m1 "$line4\n";
32.  }
33. close(m1);
34. close B;
35. ###
36. open m1,"<m1.txt");
37. open m2,">m2.txt";
38. while(<m1>){
39.  if(exists $hashc{$_})
40.    print m2 "$_\n";
41.  }
42. close(m2);
43. closes(m1);
44. ###m2 中就是取出来的 3 个的序号了
45. #
46. open LSBI ">ABC LSBL.txt";

```

```
47. open m2, "<m2.txt";
48. while(<m2>)
49. {
50.  $a=($hasha{$_}+$hashb{$_}-$hash{$_})/2;
51.  print LSBL "$a";
52. };
53. close LSBL;
54. close m2;
```

- (4) 根据公式 $LSBL = (F_{ST}(AB) + F_{ST}(AC) - F_{ST}(BC)) / 2$ 计算 LSBL 值，并且对文件进行命名。
- (5) 然后对 LSBL 进行滑动窗口计算，窗口大小为 50Kb，步长为 25Kb。
- (6) 将 F_{ST} 以及 LSBL 的结果使用 R 语言进行可视化，并且根据统计学原理基本迭代。

```
1. rm(list=ls())
2. setwd("//media//zmy//文档//LSBL")
3. mydata=read.table("w-b-o.Window.plot.txt", header=T)
4. mydata=mydata[order(as.numeric(mydata[,1]),decreasing = FALSE),]
5. colnames(mydata)=c("CHR", "BP", "P")
6. mydata=mydata[,c(1,2,3)]
7. mydata[1:10,]
8. #mydata <- mydata[which(mydata$V1 ==1),]
9. dim(mydata)
10. #source("//media//zmy//文档//LSBL//FST.source.r")
11. source("FST.source.r")
12. GAPIT.Manhattan(mydata,name.of.trait="")
```

由于一条染色体上上万的 SNP，如果都画出来的话比较消耗资源。因此我们使用 GAPIT 稀释 SNP 的方法，处理数据后做映射，然后每一个阶梯只留第一个值^[25]。

```
1. `GAPIT.Pruning` <-
2. function(values,DPP=5000000){
3.  #Object: To get index of subset that evenly distribute
4.  #Output: Index
5.  #Authors: Zhiwu Zhang  ## edit by zhkzhou@126.com
6.  # Last update: May 28, 2011
7.  #####
8.
9.  #No change if below the requirement
```

```

10. if(length(values)<=DPP)return(c(1:length(values)))
11.
12. #values= log.P.values
13.
14. values=sqrt(values) #This shift the weight a little bit to the low building
    .
15. theMin=min(values)
16. theMax=max(values)
17. range=theMax-theMin
18. interval=range/DPP
19.
20. ladder=round(values/interval)
21. ladder2=c(ladder[-1],0)
22. keep=ladder-ladder2
23. index=which(keep>0)
24.
25. return(index)
26. }#end of GAPIT.Pruning
27.
28. `GAPIT.Manhattan` <-
29. function(GI.MP = NULL, name.of.trait = "Trait",plot.type = "Genomewise", plo
    t.type2 = "Chromosomewise",
30. DPP=5000000,cutOff=0.01,band=5,seqQTN=NULL){
31. #Object: Make a Manhattan Plot
32. #Options for plot.type = "Separate_Graph_for_Each_Chromosome" and "Same_Grap
    h_for_Each_Chromosome"
33. #Input GI.MP-three columns: chorosome, bp and p
34. #Output: A pdf of the Manhattan Plot
35. #Authors: Alex Lipka, Zhiwu Zhang, and Meng Li      ## edit by zhkzhou@126.c
    om
36. # Last update: May 10, 2011
37. #####
    #####
38.
39. if(is.null(GI.MP)) return
40.
41. borrowSlot=4
42. GI.MP[,borrowSlot]=0 #Inicial as 0
43. if(!is.null(seqQTN))GI.MP[seqQTN,borrowSlot]=1
44.
45. #Eeep QTN with NA p values (set it to 1)
46. index=which(GI.MP[,borrowSlot]==1 & is.na(GI.MP[,3]))
47. GI.MP[index,3]=1
48.

```

```

49. GI.MP=matrix(as.numeric(as.matrix(GI.MP) ),nrow(GI.MP),ncol(GI.MP))
50.
51. #Remove all SNPs that do not have a chromosome, bp position and p value(NA)

52. GI.MP <- GI.MP[!is.na(GI.MP[,1]),]
53. GI.MP <- GI.MP[!is.na(GI.MP[,2]),]
54. GI.MP <- GI.MP[!is.na(GI.MP[,3]),]
55.
56. #Remove all SNPs that have P values between 0 and 1 (not na etc)
57. #GI.MP <- GI.MP[GI.MP[,3]>0,]
58. #GI.MP <- GI.MP[GI.MP[,3]<=1,]
59.
60. #Remove chr 0 and 99
61. GI.MP <- GI.MP[GI.MP[,1]!=0,]
62.
63. numMarker=nrow(GI.MP)
64. bonferroniCutOff=-log10(cutOff/numMarker)
65.
66. #Replace P the -log10 of the P-values
67. #GI.MP[,3] <- -log10(GI.MP[,3])
68.
69. #y.lim <- ceiling(max(GI.MP[,3]))
70. #y.lim = 0.2
71. #print("The max -logP vlaue is")
72. #print(y.lim)
73.
74. chm.to.analyze <- unique(GI.MP[,1])
75.
76. chm.to.analyze=chm.to.analyze[order(chm.to.analyze)]
77. numCHR= length(chm.to.analyze)
78.
79. #Chromosomewise plot
80. if(plot.type2 == "Chromosomewise")
81. {
82. #print("Manhattan plotting Chromosomewise")
83.
84. pdf(paste("Fst.", name.of.trait, ".Chromosomewise.pdf" ,sep = ""), width =
    10)
85. par(mar = c(5,5,4,3), lab = c(8,5,7))
86. for(i in 1:numCHR)
87. {
88. #Extract SBP on this chromosome
89. subset=GI.MP[GI.MP[,1]==chm.to.analyze[i],]
90.

```



```

91.   y.lim <- ceiling(max(subset[,3])) #set upper for each chr
92.   #y.lim=0.2 #set upper for each chr
93.   if(length(subset)>3){
94.     x <- as.numeric(subset[,2])/10^(6)
95.     y <- as.numeric(subset[,3])
96.   }else{
97.     x <- as.numeric(subset[2])/10^(6)
98.     y <- as.numeric(subset[3])
99.   }
100.
101.   #Prune most non important SNPs off the plots
102.   order=order(y,decreasing = TRUE)
103.   y=y[order]
104.   x=x[order]
105.
106.   index=GAPIT.Pruning(y,DPP=round(DPP/numCHR))
107.   x=x[index]
108.   y=y[index]
109.
110.   #color.vector <- subset(temp.par.data[,7], temp.par.data[,4] == i)
111.   plot(y~x,type="p", ylim=c(0,y.lim), xlim = c(min(x), max(x)), col = "navy", xlab = expression(Base~Pairs~(x10^-6)),
112.     ylab = "Fst", main = paste("Chromosome",chm.to.analyze[i],sep=" "),cex.lab=1 , pch=20, xaxs="i", yaxs="i" )
113. #abline(h=bonferroniCutOff,col="forestgreen")
114. abline(h=4,col="forestgreen")
115.   ##print("manhattan plot (chr) finished")
116. }
117. print("Manhattan-Plot.Chromosomewise finished!")
118. dev.off()
119. #print("manhattan plot on chromosome finished")
120. } #Chromosomewise plot
121.
122. #Genomewise plot
123. if(plot.type == "Genomewise")
124. {
125. #print("Manhattan plotting Genomewise")
126. #Set corlos for chromosomes
127. nchr=max(chm.to.analyze)
128. #nchr=length(chm.to.analyze) #This cause problem plotting part of chromosome
129. ncycle=ceiling(nchr/band)
130. ncolor=band*ncycle
131. palette(rainbow(ncolor+1))

```

```

132. cycle1=seq(1,nchr,by= ncycle)
133. thecolor=cycle1
134.
135. for(i in 2:ncycle){thecolor=c(thecolor,cycle1+(i-1))}
136. #print(thecolor)
137.
138.
139. #Sort by BP within CHR
140. GI.MP <- GI.MP[order(GI.MP[,2]),]
141. GI.MP <- GI.MP[order(GI.MP[,1]),]
142. color.vector <- rep(c("orangered", "navyblue"),numCHR)
143. ticks=NULL
144. lastbase=0
145.
146. #change base position to accumulatives (ticks)
147. for (i in chm.to.analyze)
148. {
149.   index=(GI.MP[,1]==i)
150.   ticks <- c(ticks, lastbase+mean(GI.MP[index,2]))
151.   GI.MP[index,2]=GI.MP[index,2]+lastbase
152.   lastbase=max(GI.MP[index,2])
153. }
154.
155. x0 <- as.numeric(GI.MP[,2])
156. y0 <- as.numeric(GI.MP[,3])
157. z0 <- as.numeric(GI.MP[,1])
158. position=order(y0,decreasing = TRUE)
159. index0=GAPIT.Pruning(y0[position],DPP=DPP)
160. index=position[index0]
161. x=x0[index]
162. y=y0[index]
163. z=z0[index]
164.
165. #Extract QTN
166. QTN=GI.MP[which(GI.MP[,borrowSlot]==1),]
167.
168. #Draw circles with same size and different thikness
169. size=1
170. ratio=5
171. base=1
172. themax=max(y)
173. themin=min(y)
174. wd=((y-themin+base)/(themax-themin+base))*size*ratio
175. s=size-wd/ratio/2

```

```

176. y.lim <- ceiling(max(GI.MP[,3]))
177. #y.lim= 20
178. #y.lim= 0.9
179. pdf(paste("Fst.", name.of.trait, ".Genomewise.pdf", sep = ""), width = 8,
      height=2.5) ##### width = 15
180. par(mar = (c(3,4,2,2)+ 0.5), mgp=c(1.6,1,0)) ##
181. par(bty="l", lwd=1.5) ## bty=l the plot is coordinate instead of box
182. mycols=rep(c("orangered", "black", "steelblue1", "green4", "navyblue"), max(
      z))## doubl dolor loop by chromosome, type = "h" is line; p is point,
183. plot(y~x, ylab=expression(italic(paste(Z, "(", "F"[st], ")", sep = "))), xaxs
      ="i", yaxs="i", type = "p", lwd=1.50, pch=20, cex.axis=0.5, cex.lab=1.0, col
      =mycols[z], axes=FALSE, lty=1, cex=0.35, cex.main=2.0, xlab="Chromosome")
      ###y ylim=c(0,y.lim),
184. mtext(paste(" ", name.of.trait, sep=""), cex=1.5, font.main =1, side=3, out
      er=TRUE, line=-1.5)
185. #lines(y, x, type = "l")
186. #Remove spacing around plotting area in r --- xaxs="i", yaxs="i"
187. if(!is.null(dim(QTN)))abline(v=QTN[,2], lty = 2, lwd=1.5, col = "grey")
188. #abline(h=0.2,col="green",, lwd=0.5)
189. title(xlab="Chromosome")
190. axis(1, at=ticks, tck=-0.01, cex.axis=1, labels=chm.to.analyze, tick=T, lwd=1
      .5, padj=-1) ## lwd: line width tick=T,
191. #axis(2, at=0:y.lim, tck=-0.01, cex.axis=1, labels=0:y.lim, lwd=1.5, padj=1)
      ## tck=-0.01 let the tck shor
192. axis(2, tck=-0.01, cex.axis=1, lwd=1.5, padj=1) ## tck=-0.01 let the tc
      k shor
193. box()
194. dev.off()
195. print("Manhattan-Plot.Genomewise finished!")
196. # mycols=rep(c("orangered", "navyblue"), max(d$CHR)) col=thecolor[z],
      rep(c("orangered", "navyblue"), z)
197. } #Genomewise plot
198.
199.
200. } #end of GAPIT.Manhattan

```

2.5.3 Pi 信号选择

(1) 我们使用 vcftools, 以 50Kb 的窗口、25Kb 的步长分别计算了两色元宝鸡以及其他家鸡的核苷酸多态性。计算时先去除所有的空集（和 LSBL 去除空集同理）。之后我们使用 perl 语言根据根据 $(\Delta \pi \text{ 或者 } \Delta Pi) = \pi_1 - \pi_2$ 计

算了二者的差值，同时计算 $\Delta \pi / \pi_1$ 的值。输出结果去除表头信息，使用管道输入文本信息^[26]。

```
1. nohup perl merge.PI.pl yuanbao_other..recode.vcf.W50S25.windowed.pi yuanbao_black..recode.vcf.W50S25.windowed.pi Other-black.merge.pi &
2. nohup perl merge.PI.pl yuanbao_white..recode.vcf.W50S25.windowed.pi yuanbao_black..recode.vcf.W50S25.windowed.pi White-black.merge.pi &
3.
4. nohup perl merge.PI.pl yuanbao_other..recode.vcf.W50S25.windowed.pi yuanbao_white..recode.vcf.W50S25.windowed.pi Other-white.merge.pi &
5. nohup perl merge.PI.pl yuanbao_black..recode.vcf.W50S25.windowed.pi yuanbao_white..recode.vcf.W50S25.windowed.pi Black-white.merge.pi &
```

其中的 merge.PI.pl 的脚本使用 perl 编写，作用就是根据公式进行计算 PI

```
1. open f1,"<$ARGV[0]";
2. open f2,"<$ARGV[1]"; ##TARGET POP
3. open f3,">$ARGV[2]";
4.
5. while(<f1>){
6.     chomp;
7.     @aa=split(/\s+/, $_);
8.     $hash{$aa[0]}{$aa[1]}=$aa[4];
9. }
10.
11. while(<f2>){
12.     chomp;
13.     $line=$_;
14.     @bb=split(/\s+/, $line);
15.     if($bb[0]=~/CHROM/){
16.
17.         print f3"$line\tother_PI\tdelta_PI\tdivid_PI\n";
18.     }
19.     elsif(exists $hash{$bb[0]}{$bb[1]})
20.     {
21.         $delta=$hash{$bb[0]}{$bb[1]}-$bb[4];
22.         $divid_PI=$hash{$bb[0]}{$bb[1]}/$bb[4];
23.         print f3"$line\t$hash{$bb[0]}{$bb[1]}\t$delta\t$divid_PI\n";
24.     }
25.
26. }
```

(2)输出的两两之间的结果进行检查,应符合以下形式,包含染色体、起始位置、结束位置、PI、差值以及标准化值等

1.	HROM		BIN_START	BIN_END	N_VARIANTS	PI	other_PI	delta_PI	divid_PI
2.	1	1	50000	264	0.00160983	0.0015864	-2.34300000000001e-05	0.985445	668176143
3.	1	25001	75000	418	0.00250773	0.00222765	-0.00028008	0.88831333516766	2
4.	1	50001	100000	391	0.00251644	0.00227812	-0.00023832	0.90529478151674	6
5.	1	75001	125000	305	0.00210274	0.00211011	7.37000000000003e-06	1.00	350495068339
6.	1	100001	150000	256	0.00171728	0.00194551	0.00022823	1.13290203111898	
7.	1	125001	175000	332	0.00209606	0.00250177	0.00040571	1.19355839050409	
8.	1	150001	200000	283	0.00150513	0.00217944	0.00067431	1.44800781327859	
9.	1	175001	225000	346	0.0015497	0.00264151	0.00109181	1.70452990901465	
10.	1	200001	250000	452	0.00238849	0.00332144	0.00093295	1.3906024308245	

2.5.4 SweeD 信号选择

SweeD 采用复合似然检验,该检测计算不同区域的单核苷酸多态性 (SNP) 的频率,通过统计学原理来对全基因组进行扫描,找到候选区域^[27]。

(1) 第一步使用 vcftools 工具,将 vcf 文件按照先亚群后染色体进行拆分,得到从三个亚群 1~28 号染色体的 vcf 小文件。这里我们采用遍历算法进行。

```
for i in {1..28}
do
# /home/zhumy/biosoft/vcftools/vcftools_0.1.13/cpp/vcftools --vcf ../yuanbao_black..
recode.vcf --chr $i --recode --out yuanbao_black.chr$i

# /home/zhumy/biosoft/vcftools/vcftools_0.1.13/cpp/vcftools --vcf ../yuanbao_white..
recode.vcf --chr $i --recode --out yuanbao_white.chr$i
```

```
/home/zhumy/biosoft/vcftools/vcftools_0.1.13/cpp/vcftools --vcf ../yuanbao_other..r
ecode.vcf --chr $i --recode --out yuanbao_other.chr$i

done
```

(2)我们使用 SweeD 的 SweeD-P 算法，以单个上述文件为输入，参数默认

```
./SweeD-P -name yuanbao_black.chr10.1KS -input yuanbao_black.chr10.recode.vcf -grid
19911 -threads 8
```

(3)在使用 perl 脚本将生成文件转化为绘图指定格式后，使用 FST 相同绘图代码进行绘制

```
open f1,"<Highdog-Lowwolf-Lowdog.Window.fst";

open f2,">Highdog-Lowwolf-Lowdog.Window.plot.txt";

while(<f1>){

    chomp;

    @aa=split(/\s+/, $_);

    if($aa[5]<=0){

        $aa[5]=0;

    }

    print f2"$aa[0]\t$aa[1]\t$aa[5]\n";

}

close f1,f2;
```

第 3 节 实验结果

3.1 FastQC 对 clean reads 测序数据进行质量评估

使用 FastQC 软件对 24 个个体的 clean reads 进行评价，所有指标均为完美，表示该测序数据可用作下一步分析

Measure	Value
Filename	good_sequence_short.txt
File type	Conventional base calls
Encoding	Illumina 1.5
Total Sequences	250000
Sequences flagged as poor quality	0
Sequence length	40
%GC	45

图 2-1 FastQC 报告的基本参数

Figure 2-1 Basic Statistics of FastQC Repor

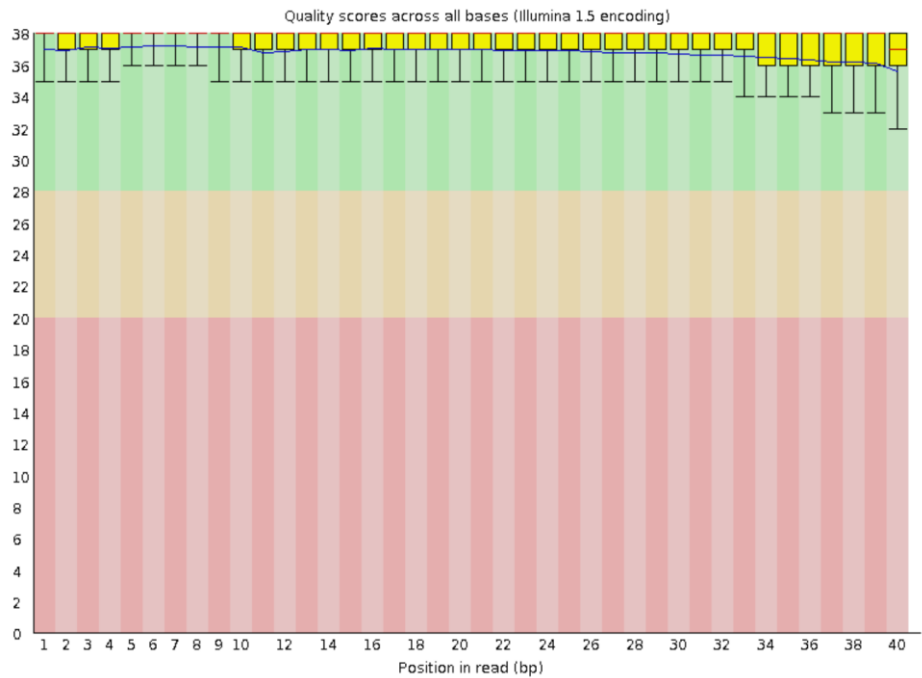


图 2-2 FastQC 报告的每个碱基质量

Figure 2-2 Per base sequence quality of FastQC Repor

3.2 ANNOVAR 注释后结果

将 ANNOVAR 注释后的 vcf 文件的注释类型使用 Excel 进行绘图，得到变异类型分布的饼图（如下图所示）：

在给定的染色体区域内，基因间突变，内含子，基因上游区域，非编码 RNA，外显子和开放阅读框中发生的 SNP 比例分别为 9112159,7584082,552045,488986,287323 和 52759。SNP 的功能注释蛋白质编码区鉴定出非同义替代的 85706 和同义替代的 200747，其中 870 个 SNP 导致终止密

码子获得或丧失。

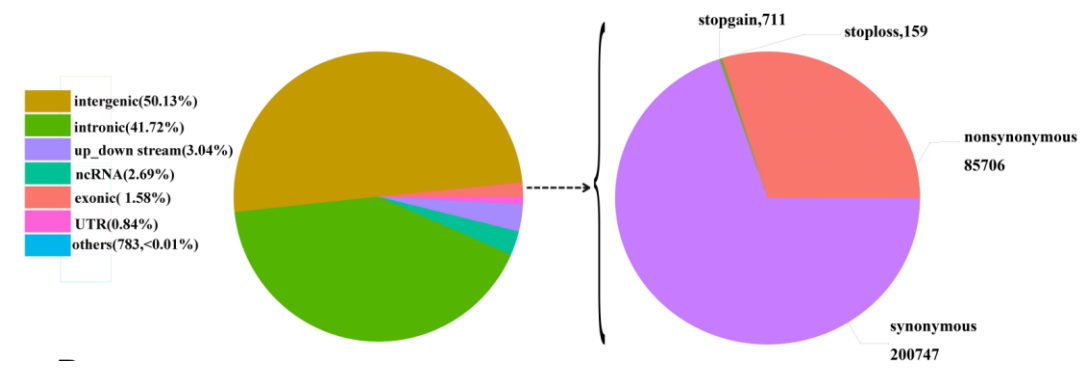


图 2-3 24 只家鸡变异类型的注释结果

Figure 2-3 Annotation results of 24 chicken variant types

3.3 基因密度

我们为了探究各个染色体与参考基因组相比的变异情况，R 语言绘制箱线图展示变异数目以及分布（如下图所示），图中可以看出大部分的常染色体都具有较高的变异程度，特别是 1~15 号染色体；性染色体变异数较少，因此在剩余部分的分析中，我们将考虑这一点。

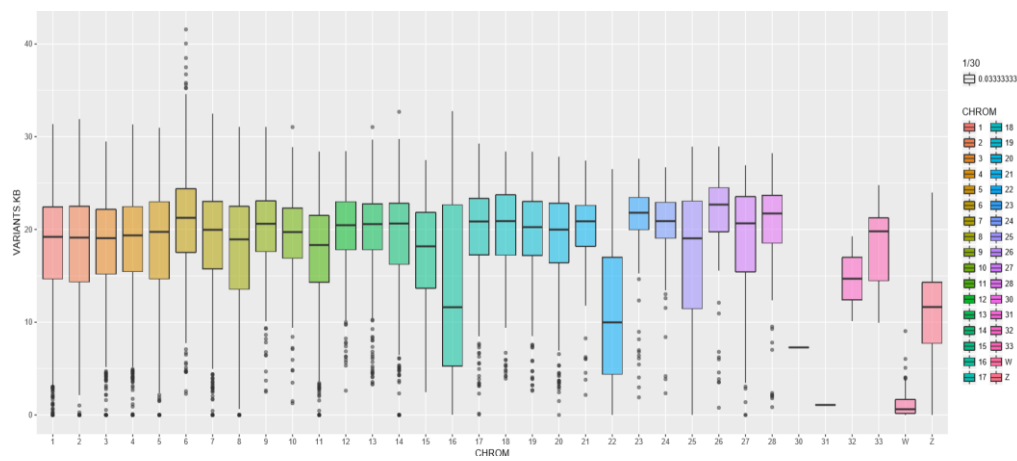


图 2-4 24 只家鸡的染色体变异情况

Figure 2-4 Chromosome variation of 24 chickens

3.4 核苷酸多态性

我们统计并输出所有个体的核苷酸多态性 PI，并进行绘图（如下图所示）图中所示，我们得到了和刚才染色体变异密度图相似的结果，变异主要集中在常染色体 1~15 之间，性染色体较少，因此在之后信号分析中我们更多向常染色体偏倚。

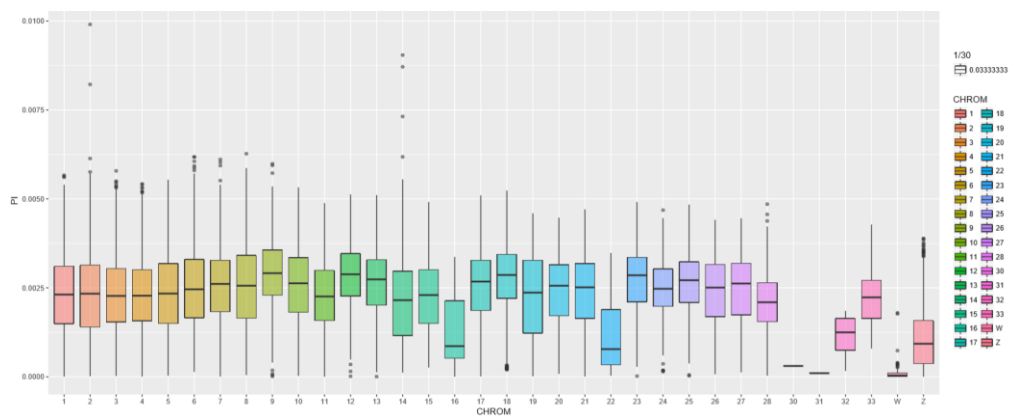


图 2-5 24 只家鸡的核苷酸多态性

Figure 2-5 Nucleotide polymorphism of 24 chickens

3.4 LSBL 分析

我们根据 LSBL 公式 ($LSBL=(F_{ST}(AB)+F_{ST}(AC)-F_{ST}(BC))/2$)，分别以元宝白鸡、元宝黑鸡、其他家鸡，以及元宝黑鸡、元宝白鸡、其他家鸡作为 ABC，命名为 B_W_O、W_B_O，如下图。可以看到，在 B_W_O LSBL、W_B_O LSBL 分析中，尽管许多表现出强烈选择信号的染色体是重合的，但具体的分布并不相同，这提示我们可能要将两者的受选择可能区域进行分开讨论，图中的较高处提示该区域信号强烈，可以作为我们定位的辅助依据。

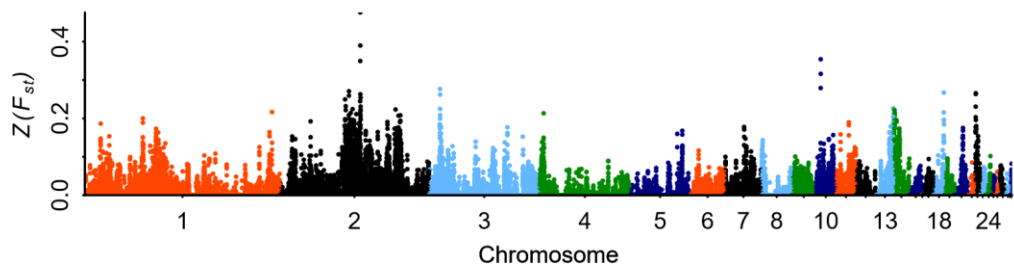


图 2-6-1 24 B_W_O LSBL 分析

Figure 2-6-1 B_W_O LSBL analysis

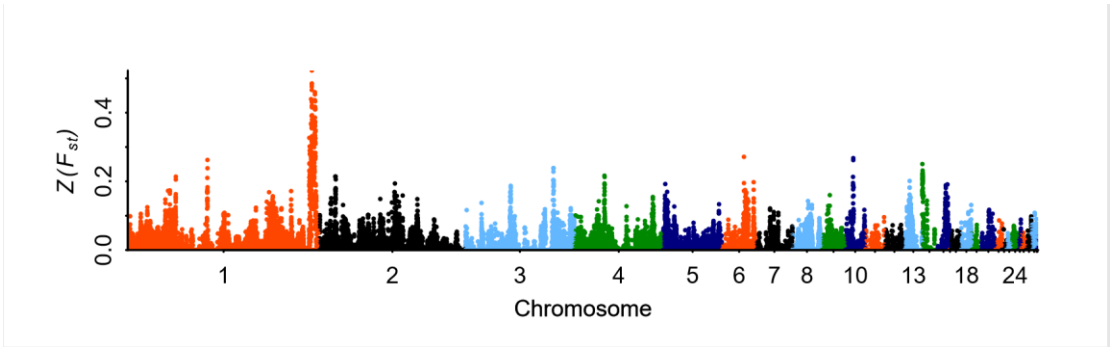


图 2-6-2 W_B_O LSBL 分析

Figure 2-6-2 W_B_O LSBL analysis

我们选中 B_W_O LSBL 分析的单个染色体区域，对变量进行变换，使之更易揭示差异，如图：

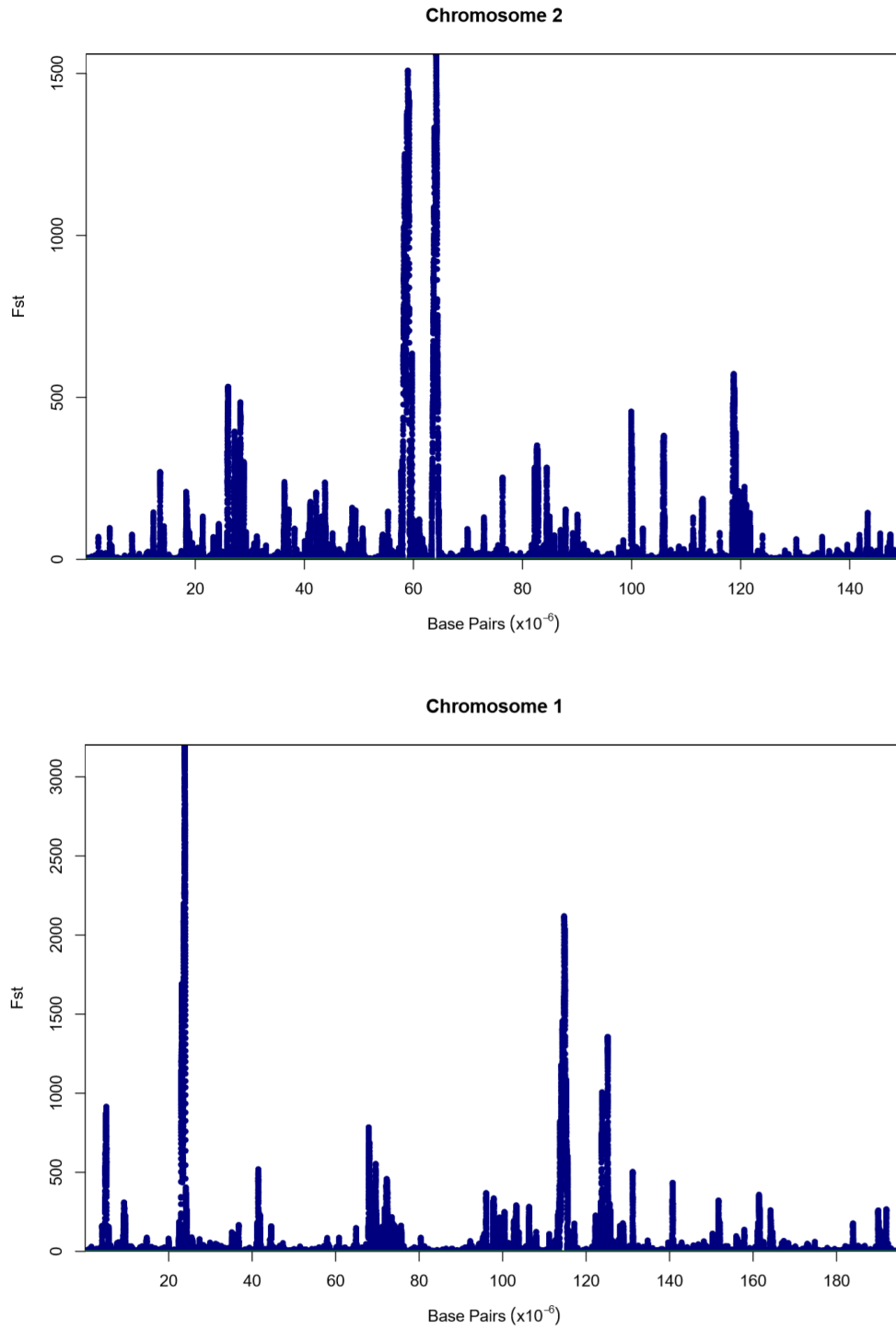


图 2-7 B_W_O LSBL 分析 1、2 染色体区域

Figure 2-7 B_W_O LSBL analysis of 1, 2 chromosome region

3.4 核苷酸多态性

我们统计并输出单个亚群的个体的核苷酸多态性 PI ，并进行绘图（如下图所示）图中所示，我们得到了和刚才染色体变异密度图相似的结果，变异主要集中在常染色体 1~15 之间，性染色体较少，因此在之后信号分析中我们更多向常染色体偏倚。

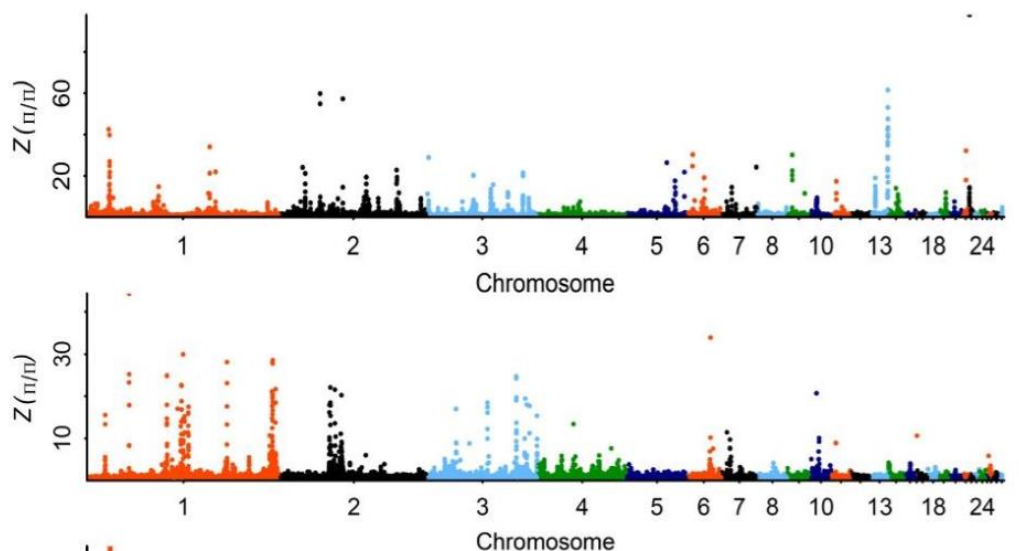


图 2-8 黑白元宝鸡的核苷酸多态性

Figure 2-8 Nucleotide polymorphism of black and white chickens

3.4 SweeD 分析

我们得到元宝黑鸡、元宝白鸡与其他家鸡的 $SweeD$ 指数作图。从图中可以看出，对于该方法，三个亚群的高选择信号区域表现在不同区域，因此结合上面我们 $LSBL$ 的分析结论，我们对于元宝黑鸡与元宝白鸡的人工选择区域分别进行分析。其中元宝黑鸡与其他亚群相比，一号染色体表现出强烈的选择信号，而在二号染色体信号较弱；元宝白鸡二号染色体表现出强烈的选择信号，而在一号染色体信号较弱。

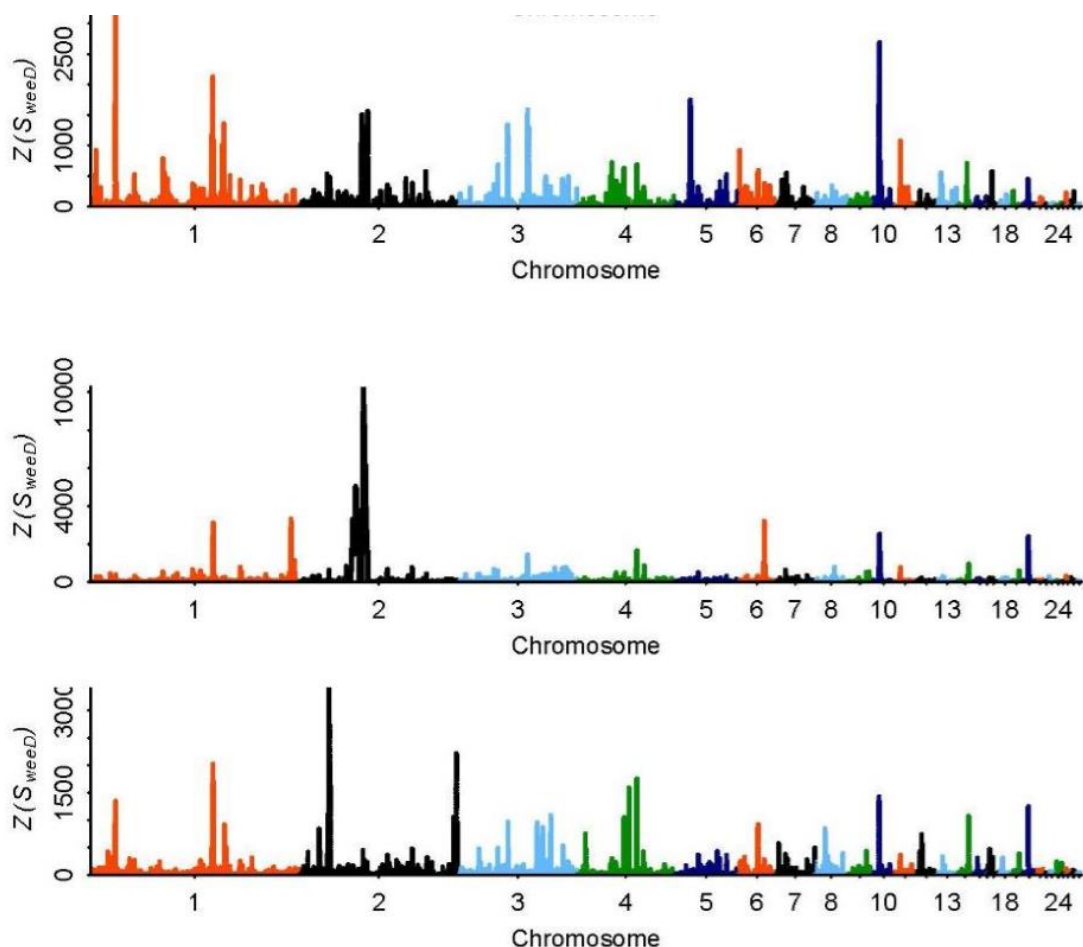


图 2-9 依次为元宝黑鸡、白鸡、其他家鸡的 SweeD 指数

Figure 2-9 the SweeD index of yuanbao black chicken, white chicken, and other chickens

第4节 讨 论

测序数据进过标准化处理后,虽然我们从图中可以看到所有的数据都完美符合标准,但是却因忽略为统计学、机器误差误差可能会隐藏部分结果。对于注释后的结果来看,大部分的突变发生在了非编码区域,编码区的突变多数也是同义替代,仅仅有少量突变会导致蛋白质的改变。我们计算了每 100kb 碱基的上的基因数目,总体比较而言,我们观察到 1、2、3 染色体 SNP 的数量最多,W 染色体上 SNP 的数量最少,并且 SNP 的数量与染色体的长度相关。但根据箱形图所示,即使考虑到染色体的长度差异带来的数目减少,W 染色体单核苷酸多样性也显著低于预期。对具有 Y 染色体的雄性染色体异性的生物的和具有 W 染色体的雌性异性的家鸡而言,无论是 Y 染色体还是 W 染色体,他们的遗传多样性明显低于预期,因此我们可以说性别决定的染色体通常具有有限的遗传多样性。

我们将所有的选择性清除分析方法进行总结，我们对比四种方法，发现各个位点的高选择信号情况不尽相同。因此我们仅能从图中推断出大致的区域位于 1、2、10 号染色体。可视化的方法仅仅能作为辅助判断的依据，我们根据作图来观察并验证选择信号的集中分布以及推断可能的选择区域，我们仍不能准确的定量确定这些区域所占的百分比、比例、信号的强弱是否在可行区间内，在后续步骤将会将方向有数据到可视化再回到数据本身，通过统计学原理验证猜想。

B_W_O LSBL、**W_B_O LSBL** 分析中我们可以看到，两者的强选择信号染色体基本吻合，并且对于 3、4、5、10、13 染色体的选择区域也基本重合，但对于 1、2 染色体的位置却不尽相同，这暗示对于单个物种与其祖先红原鸡相比较而言，相同的染色体可能是受到了同一选择，但针对不同的羽色，选择部位可能有所偏差。**SweeD** 分析中也辅助推断了该猜想，对于元宝黑羽鸡而言，1 号染色体信号最为强烈，元宝白羽鸡则是 2 号染色体。

第三章 正选择基因的注释及验证

第 1 节 实验软件

Ensemble 工具箱

Galaxy 注释工具

David 注释工具

IGV 基因组浏览器

第 2 节 实验方法

2.1 正选择基因的注释

选取 FST、Pi、LSBL、SweeD 分析中的前 1% 的值所相对应的基因组区域为候选的选择性清除的基因座位，通过 ensemble 数据库的变异预测数据库 Variant Effect Predictor (<http://asia.ensembl.org/Tools/VEP>) 的网页版工具进行注释。为了得到比较不同注释算法及数据库的结果，我们同时使用 Galaxy (<https://usegalaxy.org/>) 以及 David (<https://david.ncifcrf.gov/summary.jsp>) 进行注释。

获得的蛋白质编码基因利用 g: Profiler 进行功能注释，获得富集显著的 GO、KEGG 和 HP 功能通路。

2.2 生物学验证

按照数据库中基因的注释、通路、表达情况，从已发表文献论文、基因功能、连锁上下游通路、已验证的毛色相关通路、和人类与动物疾病五个角度来统计和评价锁定预测区域。

相关文献：被文献报道的总次数；

通路：即根据 KEGG 数据库统计基因参与的通路的数量；

功能：即根据 Go 数据库，统计基因参与的功能（生物过程）；

已验证的通路：例如与酪氨酸酶相关的黑色素合成通路；

疾病：已经有文献发表的相关白化症状以及其他物种的色素代谢疾病；

基因没有在毛色、代谢相关的通路显著富集。综合 SweeD 和 LSBL 的结果，我们发现 chr1: 187.30Mb-188.50Mb、ch1:191.45Mb191.62Mb、ch2:69.1Mb-72.9Mb 这三个区域，元宝黑羽鸡与元宝白羽鸡存在显著差异，因此推断这些区域可能参与了元宝鸡的毛色选择进化过程。

3.2 chr1: 187.30Mb-188.50M 内的 Tyr 和 Rab38 可能控制黑白元宝鸡的羽毛颜色

一号染色体上 188.25Mb-188.50Mb 区域在四种方法中拥有最高的选择信号，因此我们把它作为候选区域。根据正选择信号，该区域内有五个基因数值较高，分别是 CTSC，TYR，RAB38，gga-mir-1657 和 mGluR5a。

CTSC 与半胱氨酸型内肽酶活性和丝氨酸型内肽酶活性和分子功能相关，这可能提示该酶不参与黑色素相关代谢过程。

TYR 基因编码酪氨酸酶-色素细胞中黑色素产生的限速酶，通过影响黑色素的合成速度以及下游底物的供应影响色素合成。酪氨酸酶的酶功能若由于蛋白质氨基酸的组成改变以及构象不能正确折叠发生异常，则黑色素合成途径会被阻断，这种变异的产生会导致很多物种的白化表型，例如家鸡以及人。鸡酪氨酸酶（TYR）基因在 C 基因座处有多种突变，其中在 TYR 基因的某一段内含子中插入完整的禽逆转录病毒序列，已经作为白化的诊断依据广泛应用于临床之上。

RAB38 是 Rab 家族蛋白的成员，在 B_W_O 和 W_B_O 的 LSBL 分析中均表现出较强的选择信号。但是到目前为止，RAB38 在鸡羽毛颜色中作用尚未被报道。在其他生物中，RAB38 基因已经被证实影响黑色素的调节过程。在老鼠身上，RAB38 基因导致了褐巧克力色毛发的产生，后续试验也证实在老鼠体内，Rab38 的编码蛋白参与 TYRP1 蛋白的分选过程，同时参与真黑色素的转运和组装。RAB38 与 Tyr 基因协同，通过调控下游通路（GO: 0043473），增加组织细胞对黑色素的沉积与存储（GO: 0048770），参与构建黑素体（GO: 0042470）进而调控色素的积累。

mGluR5a（谷氨酸代谢受体 5）在神经元发育中发挥重要作用，通过行使谷氨酸受体活性，参与调节的 G 蛋白偶联受体。它同时作为神经胶质细胞外谷氨酸浓度的传感器，传递相关兴奋。

3.3 区域 ch1: 191.45-191.62, ch2: 69.1-72.9 Mb 作为候选区域可能影响毛色

我们从四种选择性清除分析的结果可以看出，ch1: 191.45-191.62 和 ch2

69.1-72.9 Mb 区域表现出较强的选择信号,提示该区域可能受到长时间的人工选择。

虽然 Ch1:191.40Mb-191.52Mb 区域含有 215 个蛋白质编码基因,但此区域尚未有文献报道有与色素调节相关的 QTL。这些基因参与各种相关的生物学过程,例如 UVRAG 参与与囊泡介导的色素等物质的逆行转运并发挥负反馈调控元件,而 PAAF1 参与构成蛋白酶复合物。而对于区域 ch2:69.1Mb-72.9Mb,基因 MC4R (黑皮质素 4 受体)参与黑皮质素受体构成和影响黑素细胞刺激激素受体活性,这一过程涉及多种生理功能的调节,包括皮肤色素沉着,类固醇生成,脂肪积累,能量稳态和外分泌腺等,但没有文献报道鸡羽毛相关的 QTL 定位于该区域。

综上所述,我们可以推测元宝鸡的黑白羽色基因的受选择基因可能位于以上位点,尽管各种推测的方法得到的结果不完全重合。

结 论

驯化过程中,家鸡的表型由于人工选择发生了巨大的改变。本实验中,基于三种亚群选择性清除分析的结果,我们找到了一系列候选基因,这些基因以及区域可能有助于我们探究家鸡毛色的驯化过程以及作为遗传育种的重要依据。

我们在 QTL 以及 GWAS 分析方法的基础上,提出了比较群体基因组学的一种新的策略,即利用 NGS 数据作为前导来推断受选择信号,这种方法更加节省时间,也更加精确。

但是由于毛色研究的复杂性、涉及通路的繁杂以及相关连锁的庞大,我们很难精确定位到单个基因。但我们的结果与前人的研究不谋而合,例如 TYR, TYRP1 和 SLC24A5 等真黑色素影响基因也包含在我们的推测区域中。结果包含一系列其他新的相关基因例如 RAB38 此前并未报道与鸡毛色相关,但许多研究表明它在小鼠黑色素的色素沉着和转运中具有特别作用,尽管其作用机制尚不清楚。

我们的结果提供了一个将家鸡毛色变异与其他动物色素沉着机制相联系的纽带,他们的机制可能在某些方面具有共性。这也为研究者探索新的基因及其功能提供了理论依据。

致谢

不管是在生活还是学习方面，四年本科学习期间，导师、朋友、同学、亲人都一直关心和帮助着我。我从大二进入实验室，已经有长达三年时间，此刻回忆这期间实验室的点点滴滴，无论是半夜最后一个离开实验室，还是在学习室读书的日子，深感学到了、成长、体会了很多。经过长时间对论文的撰写、修改、体味、琢磨，论文即将完稿，期间受到实验室的各种照顾，在此感激不尽。首先要感谢我的母校吉林大学，这是一个中国最大而且十分优秀的大学。

其次，感谢我的导师，平易近人为人谦和的赵志辉老师。在整个研究阶段，无论是平时的学习和科研，还是平时的生活，都给予了我无微不至的关怀。在做项目时，从设计方案开始，到项目制作，再到项目完成，始终指导我，到项目的制作的每一个细节。他诲人不倦的育人态度、认真踏实的工作作风、广博而专业的学识都给我留下了深刻的印象，并深深的影响我今后的人生。

同时，感谢房希碧和闫守庆老师以及实验室的各位师兄师姐们为我们的课余时间提供了更多的机会，丰富了我们的科研经历。在这个四年，不仅专业知识、动手能力得到了提高，而且丰富了我们的生活。在此，祝老师和师兄师姐们身体健康，工作顺利，事事顺心，家庭美满幸福。

再次，感谢各位老师、师兄、师姐在实践和课题上给予的指导以及生活上的帮助，他们没有架子，像我们的大哥大姐一般，一直督促和鼓励着我的学习和生活。感谢实验室各位同仁，尽管我因为年轻以及知识匮乏，惹了不少乱子，但他们依旧主动给我劝诫以及包容，在此我对他们表示衷心的感谢。

最后特别感谢我的女朋友吴瑶以及我的家人，作为我身后的精神依托，是他们使我成长努力到了现在，在此向他们感谢。

朱明远

2019年5月

参考文献

- [1]中国科学院中国动物志编辑委员会主编. 中国动物志 鸟纲 第4卷 鸡形目[M]. 1978
- [2]Carl-Johan Rubin;Michael C. Zody;Jonas Eriksson;Jennifer R. S. Meadows;Ellen Sherwood;Matthew T. Webster;Lin Jiang;Max Ingman;Ted Sharpe;Sojeong Ka;Finn Hallbäck;Francois Besnier;Örjan Carlbom;Bertrand Bed'Hom;Michele Tixier-Boichard;Per Jensen;Paul Siegel;Kerstin Lindblad-Toh;Leif Andersson. Whole-genome resequencing reveals loci under selection during chicken domestication[J]. *Nature*, 2010, Vol. 464(25): 587-593
- [3]刘益平, 朱庆, 曾凡同, 邱祥聘. 原鸡线粒体 DNA 部分序列多态性分析[J]. *畜牧兽医学报*, 2004, 第35卷(2): 134-140
- [4]陆俊贤, 贾晓旭, 唐修君, 樊艳凤, 唐梦君, 高玉时, 苏一军. 2个云南原始鸡种遗传多样性及其与红色原鸡的亲缘关系[J]. *浙江大学学报(农业与生命科学版)*, 2016, 第42卷(3): 385-390
- [5]包文斌, 束婧婷, 王存波, 张红霞, Steffen Weigend, 陈国宏, BAO Wen-bin, SHU Jing-ting, WANG Cun-bo, ZHANG Hong-xia, Steffen Weigend, CHEN Guo-hong. 中国家鸡和红色原鸡 mtDNA 控制区遗传多态性及系统进化分析[J]. *畜牧兽医学报*, 2008, (11): 1449-1459
- [6]胡日查, 满初日嘎, 赵建国, 王学梅, 李笑春, 吴科榜. 红色原鸡及其研究进展[J]. *中国家禽*, 2010, (1)
- [7]霍海龙, 霍金龙, 李大林, 苗永旺, 伍革民, 李月体. 红色原鸡群体遗传多样性[J]. *动物学杂志*, 2007, 第42卷(5): 131-135
- [8]王晓峰, 钱勇. 活禽市场规范经营的有效举措[J]. *中国禽业导刊*, 2010, (16): 5-11
- [9]Chunyan Mou;Frederique Pitel;David Gourichon;Florence Vignoles;Athanasia Tzika;Patricia Tato;Le Yu;Dave W. Burt;Bertrand Bed'hom;Michele Tixier-Boichard;Kevin J. Painter;Denis J. Headon. Cryptic Patterning of Avian Skin Confers a Developmental Facility for Loss of Neck Feathering[J]. *PLOS Biology*, 2012,
- [10]Jonas Eriksson;Greger Larson;Ulrika Gunnarsson;Bertrand Bed'hom;Michele Tixier-Boichard;Lina Strömstedt;Dominic

- Wright;Annemieke Jungerius;Addie Vereijken;Ettore Randi;Per Jensen;Leif Andersson. Identification of the yellow skin gene reveals a hybrid origin of the domestic chicken. [J]. PLoS genetics, 2008, Vol. 4: e1000010
- [11]Chen Siang Ng;Ping Wu;John Foley;Anne Foley;Merry-Lynn McDonald;Wen-Tau Juan;Chih-Jen Huang;Yu-Ting Lai;Wen-Sui Lo;Chih-Feng Chen;Suzanne M. Leal;Huanmin Zhang;Randall B. Widelitz;Pragna I. Patel;Wen-Hsiung Li;Cheng-Ming Chuong. The Chicken Frizzle Feather Is Due to an α -Keratin (KRT75) Mutation That Causes a Defective Rachis[J]. PLoS Genetics, 2012, Vol. 8(7): e1002748
- [12]Hubbard, Joanna K. 1 (Joanna.Hubbard@colorado.edu);Uy, J. Albert C. 2 ;Hauber, Mark E. 3 ;Hoekstra, Hopi E. 4 ;Safran, Rebecca J. 1. Vertebrate pigmentation: from underlying genes to adaptive function[J]. Trends in Genetics, 2010, Vol. 26(5): 231-239
- [13]郭军;曲亮;王克华;贺兴龙;. 鸡羽色性状基因定位的研究进展[J]. 中国畜牧兽医, 2012, (12): 45-50
- [14]王艳,舒鼎铭. 家禽及哺乳动物类胡萝卜素氧化酶BCM01及BCO2研究进展[J]. 中国家禽, 2015, (20): 43-47
- [15]张静,刘毅,刘安芳. 畜禽羽色候选基因ASIP和TYRP1的研究进展[J]. 中国家禽, 2015, (1): 55-58
- [16]孟浩浩,许瑞霞,代蓉,李辉,李良远,万鹏程,石国庆. 绵羊黑色素合成相关基因的研究进展[J]. 生物技术通报, 2014, (8): 34-39
- [17]徐伟,封竣淇,黄兰,蔡慧芬,罗卫星. TYR 基因研究进展[J]. 中国畜牧杂志, 2017, 第 53 卷(4): 23-27
- [18]刘小辉,周荣艳,张传生,彭永东,李祥龙. 坝上长尾鸡 TYR 基因核心启动子鉴定与单核苷酸多态性分析[J]. 农业生物技术学报, 2018, 第 26 卷(6): 959-969
- [19]刘薇. 丝羽乌骨鸡 BAC 文库的构建和黑色素相关基因 TYRP1 和 ID 的研究[D]. 中国农业大学, 2004
- [20]崔丽君,张桂贤,王雪娇,王烨,刘宇,刘伟. 鸡形目黑素皮质素受体 1 基因多态性研究[J]. 畜牧与兽医, 2014, (7): 10-16
- [21]Sakae Takeuchi;Hideyuki Suzuki;Sayoko Hirose;Masafumi Yabuuchia;Chikara Sato;Hiroaki Yamamoto;Sumio Takahashi. Molecular cloning and sequence analysis of the chick melanocortin 1-receptor gene[J]. Biochimica et Biophysica Acta: Gene Structure and Expression, 1996, Vol. 1306: 122-126
- [22]Takeuchi S;Suzuki H;Yabuuchi M;Takahashi S.. A possible involvement of melanocortin 1-receptor in regulating feather color pigmentation in the chicken[J]. Biochimica et Biophysica Acta, 1996, Vol. 1308(2): 164-168

- [23]Marie A Pointer;Nicholas I Mundy. Testing whether macroevolution follows microevolution: Are colour differences among swans (Cygnus) attributable to variation at the MC1R locus?[J]. BMC Evolutionary Biology, 2008, Vol. 8: 249
- [24]Eizirik, Eduardo 1,2 ;Yuhki, Naoya 1 ;Johnson, Warren E. 1 ;Menotti-Raymond, Marilyn 1 ;Hannah, Steven S. 3 ;O'Brien, Stephen J. 1 obrien@ncifcrf.gov. Molecular genetics and evolution of melanism in the cat family. [J]. Current Biology, 2003, Vol. 13(5): 448-453
- [25]S. Kerje 1 ;J. Lind 1 ;K. Schütz 2 ;P. Jensen 2 ;L. Andersson 1,3. Melanocortin 1-receptor (MC1R) mutations are associated with plumage colour in chicken[J]. Animal Genetics, 2003, Vol. 34(4): 241-248
- [26]Maria K. Ling 1 ;Malin C. Lagerström 1 ;Robert Fredriksson 1 ;Ronald Okimoto 2 ;Nicholas I. Mundy 3 ;Sakae Takeuchi 4 ;Helgi B. Schiöth 1. Association of feather colour with constitutively active melanocortin 1 receptors in chicken. [J]. European Journal of Biochemistry, 2003, Vol. 270(7): 1441-1449
- [27]翟正晓. 基于 RAD 简化基因组测序技术的 13 种中国地方优良鸡品种 SNPs 多态性图谱构建及群体遗传学分析[D]. 上海交通大学, 2011

