SUPPLEMENTAL MATERIAL

Supplemental Methods

Study Populations. The design of the VIRGO study has been previously described.[1] In brief, 3,501 participants hospitalized with an acute myocardial infarction, age 18 to 55 years, were enrolled between 2009 and 2012 from 103 United States and 24 Spanish hospitals using a 2:1 female-to-male enrollment design. Eligible patients had elevated cardiac biomarkers (troponin I or T or creatine kinase-myocardial band), with at least 1 biomarker >99th percentile of the upper reference limit at the recruiting center within 24 h of admission (>97% of patients had qualifying troponin levels). Additional evidence of acute myocardial ischemia was required, including either symptoms of ischemia or electrocardiogram changes indicative of new ischemia (new ST-T changes, new or presumably new left bundle branch block, or the development of pathological Q waves). Patients must have presented directly to the enrolling site or been transferred within the first 24 h of presentation. Patients who were incarcerated, did not speak English or Spanish, were unable to provide informed consent or be contacted for follow-up, developed elevated cardiac markers because of elective coronary revascularization, or had an AMI as the result of physical trauma were not eligible. Baseline patient data were collected by medical chart abstraction and standardized in-person patient interviews administered by trained personnel during the index acute myocardial infarction admission. 2,101 individuals from the United States hospital recruitment centers with available DNA and who had provided written informed consent for genetic analysis were sequenced as part of this study, of whom 2,081 (99%) were retained after application of sample quality control criteria (Table I of the online-only Data Supplement).

The design of the MESA study has been previously described and protocol available at www.mesa-nhlbi.org.[22] In brief, 6,181 men and women between the ages of 45 and 84 without prevalent cardiovascular disease were recruited between 2000-2002 from 6 United States communities. Individuals were excluded from the present study if informed consent for genetic testing had not been obtained/was withdrawn, DNA was not available for sequencing, or incident cardiovascular disease (myocardial infarction, coronary revascularization, angina, peripheral arterial disease, stroke, resuscitated cardiac arrest, death due to cardiovascular causes) through the period of last available follow-up in December 2014. 3,932 individuals underwent sequencing as part of the present study, of whom 3,761 (96%) were retained after application of sample quality control criteria (Table I of the online-only Data Supplement).

Whole Genome Sequencing. Whole genome sequencing was performed at the Broad Institute (Cambridge, MA, USA). Libraries were constructed and sequenced on the Illumina HiSeqX with the use of 151-bp paired-end reads for whole-genome sequencing. Output from Illumina software was processed by the Picard data-processing pipeline to yield BAM files containing well-calibrated, aligned reads. A sample was considered sequence complete when the mean coverage was ≥ 30x (for the MESA cohort) or ≥ 20x (for the VIRGO cohort).

Sample Quality Control. 6,033 individuals underwent whole genome sequencing, of whom 191

(3.2%) were excluded based on sequencing quality control metrics (Table I of the online-only Data Supplement). Sample exclusion criteria included:

1. DNA Contamination > 5%
2. Mean coverage < 20x
3. Sample duplicates/Identical Twins (as assessed by PI_HAT ≥ 0.95)
4. First or second degree relatives of another study participant (Kinship coefficient > 0.0884)
5. Variant Call Rate < 95%
6. Genotype/phenotype Sex Discordance or ambiguous sex ($0.5 < F_{stat} < 0.8$)

Variant Quality Control. After completion of sample level quality control, variant quality control was performed using the Hail software package (https://github.com/hail-is/hail).[3]

Variant exclusion criteria included:

1. Failure by the Genome Analysis Toolkit Variant Quality Score Recalibration metric,[4] a machine learning algorithm designed to balance sensitivity (calling genuine variants) and specificity (limit false positive variant calls)
2. Variants in low-complexity regions of the genome that preclude accurate read alignment as previously defined[5]
3. Variants in segmental duplications of the genome
4. Quality by depth score < 2 (for single nucleotide polymorphisms) or < 3 (for insertion-deletions)
5. Call rate < 95%
6. Race specific Hardy-Weinberg disequilibrium p-value < $1x10^{-6}$ in control individuals.

Race Subgroup Inference. A panel of approximately 16,000 ancestry informative markers[6] (AIMs) identified across six continental populations[7] was chosen to derive principal components (PCs) of ancestry for all samples that passed quality control. Principal component analysis was performed using EIGENSTRAT.[8]

In order to assign a race to individuals without self-reported race, a k-nearest neighbors (k-NN) classifier was applied using the first five principal components of ancestry. This analysis was done using the k-NN implementation from the Scikit-learn library in Python.[9] The classifier was built using MESA samples after removing 25 individuals with discordant self-reported race and PC ancestry as determined by visual inspection of the first two principal components of ancestry. The remaining MESA samples were split into a training set (n=2,490) and test set (n=1,246). A k-NN (k=5) classifier was built using self-reported race as the dependent variable (1: White, 2: Asian, 3: Black, 4: Hispanic) and the first five principal components of ancestry as features. The classifier had a 98.1% reclassification rate in the test set, with misclassifications generally occurring for Hispanic individuals, known to be a highly admixed population. This classifier was then applied to all 5,842 samples to generate inferred race. Inferred race and self-reported race were concordant in 5,600 of 5,831 (96%) of sample with non-missing self-reported race.

Polygenic Score Estimation. A previously derived and validated polygenic score was applied to sequenced samples using the --score option in PLINK 1.90b.[10] The algorithm multiplies the number of risk alleles at each variant by its respective weight and sums across variants to generate a polygenic score for each individual. In total, 6,286,512 of 6,630,150 variants (94.8%) were available. Missing genotypes were imputed to the ancestry-specific mean allele frequency.

Supplemental Code

Supplementary Code I. Wrapper Python script to extract and score samples in a variant call format (.vcf) file

```
############################################################
# CODE TO EXTRACT AND SCORE SAMPLES WITH POLYGENIC SCORES #
############################################################
# AUTHOR: MARK CHAFFIN
# DATE COMPILED: 07-23-2018
# VERSION: 0.0.1
# NOTE: PROGRAM IS BETA, PROCEED WITH CAUTION
########################
# REQUIRED TOOLS/DATA #
########################
1) python (version 2.7, https://www.python.org/download/releases/2.7/)
      pandas (version >= 0.17.0) -- tested with version 0.23.0
      numpy -- tested with version 1.14.3
      pysam (http://pysam.readthedocs.io/en/latest/) -- tested with version 0.9.1.4
2) vcf_prep_for_score_V2.py
3) plink (plink1.9 for hard-called or sequenced data https://www.cog-
genomics.org/plink2; plink2 if dosage data from imputation https://www.cog-
genomics.org/plink/2.0/)
4) polygenic score file (CoronaryArteryDisease_PRS_LDpred_rho0.001_v3.txt)
5) input vcf.gz file (with accompanying tabix-index) to score
#############################
# EXTRACT VARIANTS FROM VCF #
#############################
#look at the documentation of the vcf_prep_for_score.py script
python vcf_prep_for_score_V2.py --help
usage: vcf_prep_for_score_V2.py [-h] --vcf VCF --score SCORE [--header]
                                [--out OUT]
optional arguments:
  -h, --help     show this help message and exit
  --vcf VCF      Path to the vcf.gz to convert. Should be of typical VCF
                 format, in gzipped form and must have tabix-index file in
                 same location. Multi-allelic variants must have separate rows
                 for each variant.
  --score SCORE  Score file for score, formatted as input for PLINK. First 3
                 columns should be Variant, Effect Allele, and Effect Weight,
                 respectively. Assumes the file is tab-delimited. The variant
                 ID should be denoted as chromosome:position:a1:a2, where the
                 order of a1 and a2 does not matter. Any commented lines at
                 the top with "#" are ignored.
  --header       Flag to indicate that the score file has a header row
                 (ignoring lines beginning with "#").
  --out OUT      The name of the output bcf file. Default is out.bcf in
                 current directory.
#prep the vcf.gz file for input to plink
python vcf_prep_for_score_V2.py \
--vcf /path/to/your_vcf.vcf.gz \
--score CoronaryArteryDisease_PRS_LDpred_rho0.001_v3.txt \
--header \
--out /path/to/output_bcf.bcf
#this script may be slow depending on the size of the VCF (# of variants and # of
samples)
#we recommend running this command by chromosome, or smaller chunks (dispatching to a
job scheduler)
#note: this assumes that multi-allelic variants have been split into separate rows in
the input VCF
###################
```

```
# SCORING SAMPLES #
###################
#remove the header from the score file for use in plink (header rows begin with #)
sed '/^#/ d' CoronaryArteryDisease_PRS_LDpred_rho0.001_v3.txt >
CoronaryArteryDisease_PRS_LDpred_rho0.001_v3_forPlink.txt
#apply the default scoring procedure in plink. missing data is imputed to the mean
value for a particular genotype.
#we therefore recommend scoring samples by ancestry group (European Samples in the
xample below) to obtain the most accurate results.
#more details on plink options for scoring can be found at https://www.cog-
genomics.org/plink/1.9/score
plink --bcf /path/to/output_bcf.bcf \
--double-id \
--allow-no-sex \
--keep EuropeanSamples.txt \
--score CoronaryArteryDisease_PRS_LDpred_rho0.001_v3_forPlink.txt header sum \
--out /path/to/output_score
#Note: if doing scoring by chromosome/chunk, take the sum of the SCORESUM column
across all chromosomes/chunks to obtain final scores for samples
#if scoring imputed data by allelic dosage, consider using plink2 specifying dosage as
"GP" (posterior probability per genotype), "DS" (Minimac3-style dosage),
#or "HDS" (Minimac4-style DS+HDS phred dosage). An example with dosage=DS is shown
below. Note: PLINK2 hasn't implemented BCF compatability so we first convert
#to a VCF in the below example. This uses the tool bcftools
(https://samtools.github.io/bcftools/bcftools.html)
bcftools convert -o /path/to/output_bcf.vcf.gz -Oz /path/to/output_bcf.bcf
plink2 --vcf /path/to/output_bcf.vcf.gz dosage=DS \
--double-id \
--score CoronaryArteryDisease_PRS_LDpred_rho0.001_v3_forPlink.txt \
--out /path/to/output_score
```

Supplementary Code II. Python script to extract and score samples in a variant call format (.vcf) file

```
#!/usr/bin/env python
# Author: Mark Chaffin
# Date: 23 July 2018

"""
This script prepares a vcf.gz file for scoring via PLINK. Takes as input a vcf.gz and
score file and harmonizes the two. The vcf.gz should
follow standard vcf format, and the score file should be a typical PLINK score file
with first 3 columns Variant, Effect Allele, and Effect Weight.
Variants in the score file are expected to be denoted as chromosome:position:a1:a2.
Order of alleles a1 and a2 do not matter in the score file as
both orders will be checked for in the vcf.gz. Output is saved as a .bcf file for
quicker computation downstream with plink.

Usage:
python vcf_prep_for_score_V2.py --vcf=VCF --score=SCORE --header --out=OUT

Ex)
python vcf_prep_for_score_V2.py --vcf /path/to/yourvcf.vcf.gz \
--score CoronaryArteryDisease_PRS_LDpred_rho0.001_v3.txt \
--header \
--out /path/to/output.bcf

VCF: The path and name of the vcf.gz file to prepare
SCORE: The score file in traditional PLINK format. The first 3 columns should be
variant, effect allele, and effect weight. Any rows beginning with #
       in the header of the file will be ignored. If a header is present, use the --
header flag, otherwise i
OUT: The path and name of the output file of interest (to be saved as a .bcf)

"""
from __future__ import division
import pandas as pd
import numpy as np
import os
import re
import sys
import gzip
import time, sys, traceback, argparse
from pysam import VariantFile


pd.options.mode.chained_assignment = None
np.set_printoptions(precision=3)

__version__ = '0.0.1'

def extract_vcf(vcf, score, out, header):
    '''Stream the vcf, keeping only rows of interest and re-naming variant IDs'''
    if header:
        scorefile = pd.read_csv(score, sep='\t', comment='#')
        if (scorefile.columns[1] in ['A','C','G','T']):
            raise ValueError('Looks like the score file does not have a header? Try
removing the --header flag')
    else:
        scorefile = pd.read_csv(score, sep='\t', comment='#', header=None)
        if (not scorefile.iloc[0].tolist()[1] in ['A','C','G','T']):
            raise ValueError('Looks like the score file has a header? Try adding the -
-header flag')
```

```
    #store the variants in dataframe by chromosome
    vardict = 11
    scorefile['chr'] = [int(x.split(':')[0]) for x in scorefile[scorefile.columns[0]]]
    for i in range(1,23,1):
        goodvariants = None
        goodvariants =
set(scorefile[scorefile['chr']==i][scorefile.columns[0]].tolist())
        print(str(len(goodvariants)) + " variants on chromosome " + str(i) + " in
score file")
        vardict[i] = goodvariants
    #use pysam to stream through the VCF
    vcf_in = VariantFile(vcf)  # auto-detect input format
    vcf_out = VariantFile(out, 'wb', header=vcf_in.header)
    start = time.time()
    ct=0
    for en,rec in enumerate(vcf_in.fetch()):
        if en % 10000 == 0:
            end = time.time()
            print(str(en) + ' rows of VCF parsed; ' + str(ct) + ' variants matched to
score file; time elapsed ' + str((end - start)))
        chrom = rec.chrom
        id1 = rec.chrom + ":" + str(rec.pos) + ":" + rec.ref + ":" + rec.alts[0]
        id2 = rec.chrom + ":" + str(rec.pos) + ":" + rec.alts[0] + ":" + rec.ref
        if id1 in vardict[int(chrom)]:
            rec.id = id1
            vcf_out.write(rec)
            ct+=1
        elif id2 in vardict[int(chrom)]:
            rec.id = id2
            vcf_out.write(rec)
            ct+=1

parser = argparse.ArgumentParser()
parser.add_argument('--vcf', type=str, help='Path to the vcf.gz to convert. Should be
of typical VCF format, in gzipped form and must have tabix-index file in same
location. Multi-allelic variants must have separate rows for each variant.',
required=True)
parser.add_argument('--score', type=str, help='Score file for score, formatted as
input for PLINK. First 3 columns should be Variant, Effect Allele, and Effect Weight,
respectively. Assumes the file is tab-delimited. The variant ID should be denoted as
chromosome:position:a1:a2, where the order of a1 and a2 does not matter. Any commented
lines at the top with \"#\" are ignored.', required=True)
parser.add_argument('--header', help='Flag to indicate that the score file has a
header row (ignoring lines beginning with \"#\").', action='store_true')
parser.add_argument('--out', type=str, help='The name of the output bcf file. Default
is out.bcf in current directory.', default="out.bcf")

if __name__ == "__main__":
    #Check of the input to make sure everything necessary is provided
    args = parser.parse_args()
    if args.vcf is None:
        raise ValueError('--vcf is required')
    if args.score is None:
        raise ValueError('--score is required')
    if args.out is None:
        raise ValueError('--out is required')

    #Check if your variant is present in the two data sources
    print('#########################################################\n' + \
          '#      RUNNING POLYGENIC SCORE PREPARATION FILE      #\n' + \
          '#    Note: This is a beta script, use at own risk    #\n' + \
          '#########################################################')
    extract_vcf(args.vcf, args.score, args.out, args.header)
```

Supplementary Code III. R Code to adjust polygenic score for genetic ancestry

```
# Author: Amit Khera
# Date: 23 July 2018

"""
#This code, run in R, enables correction of raw polygenic score for genetic ancestry.

#pheno: R data.frame
#score: Vector in data.frame with polygenic scores
#mi:   Variable denoting myocardial infarction patients (=1) or controls (=0)
#pc1-4 Variables denoting values for first four principal components of ancestry
"""

#Use a linear regression model to predict the polygenic score based on the first four
principal components of ancestry among control individuals of the dataset.#

pcmod=lm(score~pc1+pc2+pc3+pc4,data=pheno[which(pheno$mi==0),])

#Use this model to predict the polygenic score in the entire dataset based only on
genetic ancestry (first four principal components of ancestry).#

pheno$predictedscore=predict(pcmod,pheno)

#Subtract this predicted score from the raw polygenic score observed to calculate a
residualized score for subsequent analysis#

pheno$scoreresid = pheno$score-pheno$predicted score
```

Supplemental Table I. Sample Quality Control Criteria

	Thresholds	Controls	MI Patients	Total
Initial Sample Size		3932	2101	6,033
Contamination	> 5.0 %	19	3	22
Raw Mean Coverage	< 20X	1	2	3
Duplicates/Twins	PI-Hat $\geq$ 0.95	2	10	12
$1^{st}/2^{nd}$ Degree Relatives	0.0884 < Kinship Coefficient < 0.354	148	2	150
Post-QC Call Rate	< 95%	0	3	3
Sex Check	$0.5 < F_{stat} < 0.8$	1	0	1
Total Patients		3761	2081	5842

QC – quality control

Supplemental Table II. Familial Hypercholesterolemia Mutation Prevalence and Impact in Patients with Early-onset Myocardial Infarction and Controls

Mutation class	**N Carriers (%) Among 2,081 Early-Onset MI Participants**	**Mean LDL cholesterol, mg/dl**	**Impact on LDL cholesterol, mg/dl (95%CI)**
Loss-of-function *LDLR*	8 (0.4%)	297	+ 174 (137 – 212)
ClinVar Pathogenic	12 (0.6%)	192	+ 75 (46 – 105)
Rare *LDLR* missense	16 (0.8%)	174	+ 49 (26 – 73)
Combined	36 (1.7%)	202	+ 82 (65 – 99)
Mutation class	**N Carriers (%) Among 3,761 Controls**	**Mean LDL cholesterol, mg/dl**	**Impact on LDL cholesterol, mg/dl (95%CI)**
Loss-of-function *LDLR*	1 (0.03%)	186	+ 58 (-7 – +123)
ClinVar Pathogenic	7 (0.2%)	148	+25 (-1 – +52)
Rare *LDLR* missense	15 (0.4%)	147	+ 26 (9 – 43)
Combined	23 (0.6%)	149	+26 (13 – 40)

Low density lipoprotein receptor (*LDLR*) loss-of-function mutations included those that predicted to inactivate protein function due to premature truncation, frameshift, splice-site, or structural deoxyribonucleic acid (DNA) variation. ClinVar pathogenic variants include those previously annotated as either 'pathogenic' or 'likely pathogenic' in an online genetics database. Rare *LDLR* missense mutations included those with allele frequency < 1% annotated as damaging or possible damaging by each of five computer prediction algorithms (likelihood ratio test [LRT] score, MutationTaster, PolyPhen-2 HumDiv, PolyPhen-2 HumVar, and Sorting Intolerant From Tolerant [SIFT]). Impact on LDL cholesterol was assessed via comparison to a reference group of familial hypercholesterolemia mutation noncarriers in a linear regression model adjusted for age, sex, and principal components of ancestry.

LDL – low density lipoprotein

Supplemental Table III. Familial Hypercholesterolemia Mutations in Early-Onset Myocardial Infarction Patients and Controls

Variant*	Gene	Consequence	Variant Type	Amino Acid or cDNA Change	Allele Freq GnoMAD†	N Carriers of 2081 Patient	N Carriers of 3761 Controls
19:11211016_C/T	*LDLR*	missense_variant	Predicted Damaging	p.Thr62Met	1.1E-04	0	1
19:11213349_C/T	*LDLR*	missense_variant	Predicted Damaging	p.Thr67Ile	1.1E-05	1	0
19:11213408_T/G	*LDLR*	missense_variant	ClinVar Pathogenic	p.Trp87Gly	2.9E-05	1	0
19:11213448_A/T	*LDLR*	missense_variant	Predicted Damaging	p.Asp100Val	3.2E-05	1	0
19:11215980_A/C	*LDLR*	missense_variant	Predicted Damaging	p.Asp133Ala		0	1
19:11216076_G/T	*LDLR*	missense_variant	Predicted Damaging	p.Trp165Leu	3.2E-05	1	0
19:11216133_G/A	*LDLR*	missense_variant	ClinVar Pathogenic	p.Cys184Tyr	9.7E-05	3	0
19:11216172_G/A	*LDLR*	missense_variant	ClinVar Pathogenic	p.Cys197Tyr		1	1
19:11216264_G/T	*LDLR*	stop_gained	Loss-of-function	p.Glu228Ter	1.1E-05	1	0
19:11217306_C/G	*LDLR*	missense_variant	Predicted Damaging	p.Gln254Glu	1.4E-05	0	2
19:11218069_GA/G	*LDLR*	frameshift_variant	Loss-of-function	p.Thr274HisfsTer96	7.2E-06	1	0
19:11218160_G/A	*LDLR*	missense_variant	ClinVar Pathogenic	p.Asp304Asn	1.1E-05	1	0
19:11221334_A/G	*LDLR*	missense_variant	Predicted Damaging	p.Asn316Ser	1.8E-05	1	0
19:11221390_G/A	*LDLR*	missense_variant	Predicted Damaging	p.Gly335Ser	2.9E-05	0	2
19:11221414_G/A	*LDLR*	missense_variant	Predicted Damaging	p.Gly343Ser	2.5E-05	1	0
19:11221440_ATGCGAAGG/A	*LDLR*	splice_donor_variant	Loss-of-function	c.1056_1060+3delCGAAGGTG	3.2E-05	1	0
19:11222190_A/T	*LDLR*	missense_variant	Predicted Damaging	p.Asp354Val	6.5E-05	1	0
19:11222232_G/A	*LDLR*	missense_variant	Predicted Damaging	p.Cys368Tyr	1.4E-05	1	0
19:11222247_G/A	*LDLR*	missense_variant	Predicted Damaging	p.Gly373Asp	3.2E-05	1	0
19:11222264_T/C	*LDLR*	missense_variant	Predicted Damaging	p.Cys379Arg	3.2E-05	1	0
19:11223962_G/A	*LDLR*	missense_variant	Predicted Damaging	p.Ala399Thr	2.0E-05	1	0
19:11223968_C/G	*LDLR*	missense_variant	ClinVar Pathogenic	p.Leu401Val	2.2E-05	2	1

19:11223983_C/T	*LDLR*	missense_variant	ClinVar Pathogenic	p.Arg406Trp	1.8E-05	1	0
19:11224005_C/T	*LDLR*	missense_variant	Predicted Damaging	p.Thr413Met	3.7E-05	0	1
19:11224014_G/A	*LDLR*	missense_variant	Predicted Damaging	p.Arg416Gln	1.8E-05	1	0
19:11224102_C/G	*LDLR*	missense_variant	Predicted Damaging	p.Asp445Glu	3.2E-05	1	0
19:11224109_C/T	*LDLR*	stop_gained	Loss-of-function	p.Gln448Ter	7.2E-06	1	0
19:11224296_G/A	*LDLR*	missense_variant	Predicted Damaging	p.Asp482Asn	4.0E-05	1	0
19:11224326_G/C	*LDLR*	missense_variant	Predicted Damaging	p.Asp492His	7.2E-06	1	0
19:11224327_ACT/A	*LDLR*	frameshift_variant	Loss-of-function	p.Ser493CysfsTer42	3.2E-05	1	0
19:11224419_G/C	*LDLR*	missense_variant	Predicted Damaging	p.Val523Leu	4.1E-06	0	1
19:11224428_C/T	*LDLR*	missense_variant	Predicted Damaging	p.Pro526Ser	1.1E-05	1	0
19:11226775_T/C	*LDLR*	missense_variant	Predicted Damaging	p.Met531Thr		0	1
19:11227576_C/T	*LDLR*	missense_variant	ClinVar Pathogenic	p.His583Tyr	1.0E-04	0	2
19:11227590_C/G	*LDLR*	missense_variant	Predicted Damaging	p.Ser587Arg	1.6E-05	0	1
19:11227604_G/A	*LDLR*	missense_variant	ClinVar Pathogenic	p.Gly592Glu	5.8E-05	3	1
19:11227612_C/T	*LDLR*	missense_variant	Predicted Damaging	p.Arg595Trp	7.2E-06	1	0
19:11227664_C/T	*LDLR*	missense_variant	Predicted Damaging	p.Ala612Val	7.2E-06	0	1
19:11230767_G/A	*LDLR*	splice_acceptor_variant	Loss-of-function	c.1846-1G>A	3.2E-05	1	0
19:11230819_C/T	*LDLR*	missense_variant	ClinVar Pathogenic	p.Arg633Cys	1.2E-05	0	1
19:11231084_G/C	*LDLR*	missense_variant	Predicted Damaging	p.Gly676Arg	4.1E-06	0	1
19:11231136_A/T	*LDLR*	missense_variant	Predicted Damaging	p.Lys693Met		0	1
19:11231154_C/T	*LDLR*	missense_variant	Predicted Damaging	p.Pro699Leu	4.0E-05	0	1
19:11233876_G/T	*LDLR*	stop_gained	Loss-of-function	p.Glu723Ter	3.2E-05	1	0
19:11240329_G/A	*LDLR*	missense_variant	Predicted Damaging	p.Gly844Ser		0	1
7.9 kilobase deletion	*LDLR*	structural_variant	Loss-of-function	Four-exon deletion		1	0
1.7 kilobase deletion	*LDLR*	structural_variant	Loss-of-function	One-exon deletion		0	1

2:21229161_G/A	*APOB*	missense_variant	ClinVar Pathogenic	p.Arg3527Trp	1.4E-04	0	1

* Variant is described based on' chromosome:position:reference allele:alternate allele' formatting, with chromosome positions based on the hg19 genome assembly.
† Allele frequency derived from the gnomAD Genome Aggregation Database, a publicly available population allele frequency database of up to 138,362 individuals (http://gnomad.broadinstitute.org)

cDNA – complementary deoxyribonucleic acid; *LDLR* – LDL receptor gene; *APOB* – Apolipoprotein B gene

Supplemental Table IV. Baseline Characteristics of Patients with Early-onset Myocardial Infarction and Controls according to Presence of High Polygenic Score

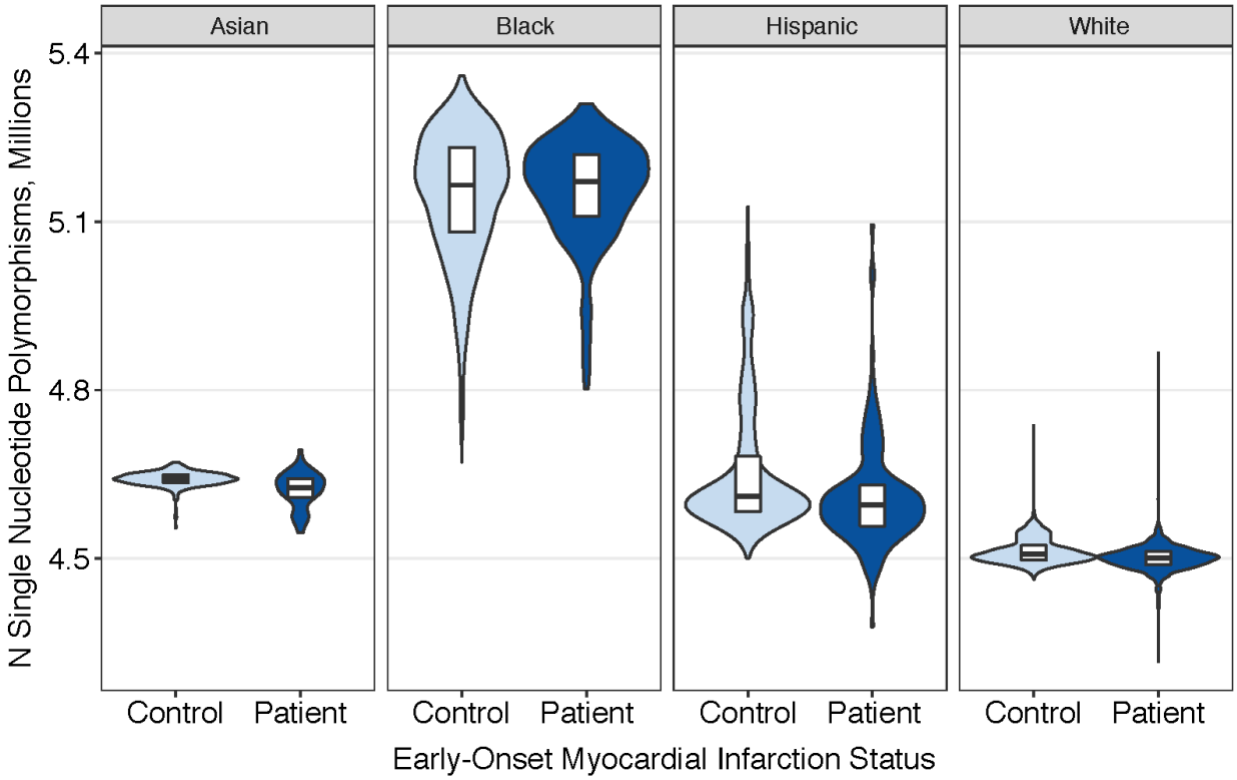
	Remainder of the distribution (N = 1,722)	**High polygenic score (N = 359)**
Race, N (%)		
White	1,252 (73%)	285 (79%)
Black	301 (17%)	35 (10%)
Hispanic	136 (8%)	32 (9%)
Asian	33 (2%)	7 (2%)
Male sex, N (%)	584 (34%)	125 (35%)
Age, years; Mean (SD)	47.6 (5.9)	47.7 (5.7)
Hypertension, N (%)	1,099 (64%)	246 (69%)
Diabetes, N (%)	599 (35%)	136 (38%)
Current Smoking, N (%)	862 (50%)	193 (54%)
Statin Use, N (%)	460 (27%)	115 (32%)
Lipid Levels, mg/dl		
LDL Cholesterol; Mean (SD)*	124 (48)	132 (52)
HDL Cholesterol; Mean (SD)	41 (14)	39 (13)
Triglycerides; Median (Q1,Q3)	133 (91 – 205)	155 (104 – 220)

N – number; SD- standard deviation; LDL – low density lipoprotein; HDL – high density lipoprotein; Q1 – quartile 1; Q3 – quartile 3

* In order to estimate untreated values for LDL, measured values for those reporting use of statin medications were divided by 0.7.

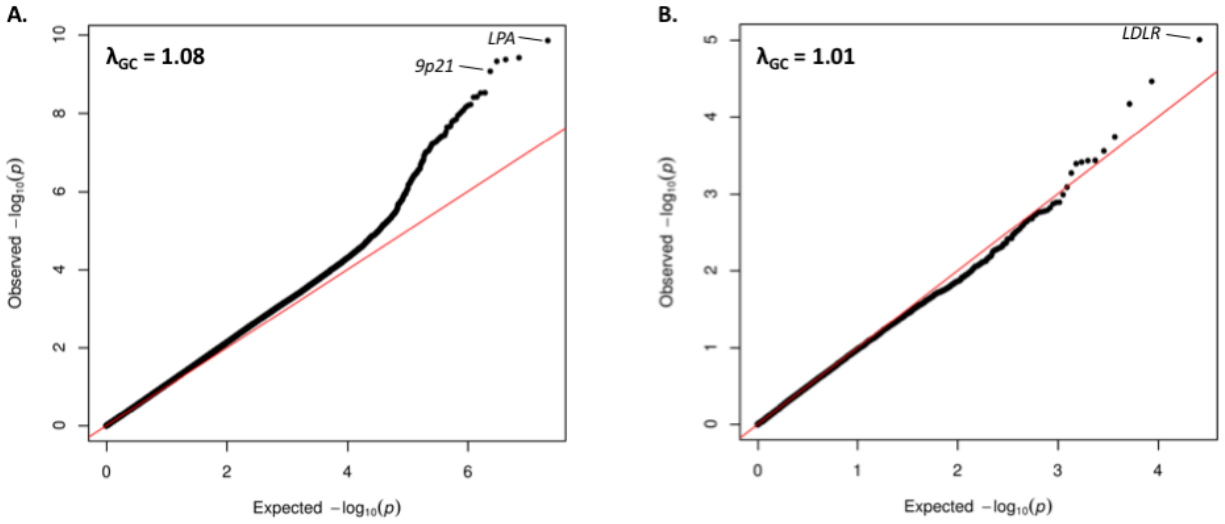
Supplemental Figure I. Single nucleotide polymorphisms in early-onset myocardial infarction patients versus controls

As expected based on mean whole genome sequencing target coverage of > 30x for the MultiEthnic Study of Atherosclerosis (MESA) cohort and > 20x for the Variation in Recovery: Role of Gender on Outcomes of Young AMI Patients (VIRGO) study, mean depth was slightly lower in myocardial infarction patients as compared to controls (29.5 versus 33.2 respectively). Despite this, the number of single nucleotide polymorphisms was similar between patients and controls in a race-stratified analysis, as displayed in violin plots. Within the white boxplot insets, the horizontal line in each box indicates the median score, and the top and bottom of the boxes indicate the 75th and 25th percentiles, respectively.



Supplemental Figure II. Common variant and gene-based rare variant burden association test quantile-quantile plots

A single variant analysis tested the relationship between all common (allele frequency ≥ 1%) single nucleotide polymorphisms (n = 10,635,039) with early-onset myocardial infarction patient status, using logistic regression adjusted for the first four principal components of ancestry. A quantile-quantile plot comparing observed versus expected *P* values is displayed in in Panel **A**, corresponding to a genomic control factor (λ_{GC}) of 1.08. Only variants at the previously well-validated lipoprotein(a) (*LPA; P* = 1.4x10^{-10}) and *9p21* (*P* = 8.4x10^{-10}) reached a *P* values of < 5 x 10^{-9}, the recommended threshold for whole genome sequencing analysis.[11] In order to assess rare variants (allele frequency < 1%), which do not occur with sufficient frequency to test in isolation, we aggregated variants predicted to cause loss-of-function, missense variants predicted to be damaging by each of five computational prediction algorithms, and those annotated as pathogenic in ClinVar.[12] 13,017 genes had at least 2 variants and at least 5 carriers in our population to enable testing in a logistic regression model adjusted for the first four principal components of ancestry. In Panel **B**, we display the quantile-quantile plot comparing observed versus expected *P* values for this analysis. Although no gene reached recommended levels for exome-wide statistical significance (*P* < 2.5 x 10^{-6}), the top signal was for the low-density lipoprotein receptor gene (*LDLR*; p = 1.0 x 10^{-5}), in which rare damaging variants were associated with a 3.87-fold (95%CI 2.12—7.06) increased risk for early-onset myocardial infarction, consistent with our previous analysis using entirely independent study cohorts.[13]



Supplemental References

1. Lichtman JH, Lorenze NP, D'Onofrio G, Spertus JA, Lindau ST, Morgan TM, Herrin J, Bueno H, Mattera JA, Ridker PM and Krumholz HM. Variation in recovery: Role of gender on outcomes of young AMI patients (VIRGO) study design. *Circ Cardiovasc Qual Outcomes*. 2010;3:684-693.
2. Bild DE, Bluemke DA, Burke GL, Detrano R, Diez Roux AV, Folsom AR, Greenland P, Jacob DR, Jr., Kronmal R, Liu K, Nelson JC, O'Leary D, Saad MF, Shea S, Szklo M and Tracy RP. Multi-Ethnic Study of Atherosclerosis: objectives and design. *Am J Epidemiol*. 2002;156:871-881.
3. Ganna A, Satterstrom FK, Zekavat SM, Das I, Kurki MI, Churchhouse C, Alfoldi J, Martin AR, Havulinna AS, Byrnes A, Thompson WK, Nielsen PR, Karczewski KJ, Saarentaus E, Rivas MA, Gupta N, Pietilainen O, Emdin CA, Lescai F, Bybjerg-Grauholm J, Flannick J, Mercader JM, Udler M, Laakso M, Salomaa V, Hultman C, Ripatti S, Hamalainen E, Moilanen JS, Korkko J, Kuismin O, Nordentoft M, Hougaard DM, Mors O, Werge T, Mortensen PB, MacArthur D, Daly MJ, Sullivan PF, Locke AE, Palotie A, Borglum AD, Kathiresan S and Neale BM. Quantifying the Impact of Rare and Ultra-rare Coding Variation across the Phenotypic Spectrum. *Am J Hum Genet*. 2018;102:1204-1211.
4. McKenna A, Hanna M, Banks E, Sivachenko A, Cibulskis K, Kernytsky A, Garimella K, Altshuler D, Gabriel S, Daly M and DePristo MA. The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data. *Genome Res*. 2010;20:1297-1303.
5. Li H. Toward better understanding of artifacts in variant calling from high-coverage samples. *Bioinformatics*. 2014;30:2843-2851.
6. Hoggart CJ, Parra EJ, Shriver MD, Bonilla C, Kittles RA, Clayton DG and McKeigue PM. Control of confounding of genetic associations in stratified populations. *Am J Hum Genet*. 2003;72:1492-1504.
7. Libiger O and Schork NJ. A Method for Inferring an Individual's Genetic Ancestry and Degree of Admixture Associated with Six Major Continental Populations. *Front Genet*. 2012;3:322.
8. Price AL, Patterson NJ, Plenge RM, Weinblatt ME, Shadick NA and Reich D. Principal components analysis corrects for stratification in genome-wide association studies. *Nat Genet*. 2006;38:904-909.
9. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M and Duchesnay E. Scikit-learn: Machine Learning in Python. *J Mach Learn Res*. 2011;12:2825-2830.
10. Chang CC, Chow CC, Tellier LC, Vattikuti S, Purcell SM and Lee JJ. Second-generation PLINK: rising to the challenge of larger and richer datasets. *Gigascience*. 2015;4:7.
11. Pulit SL, de With SA and de Bakker PI. Resetting the bar: Statistical significance in whole-genome sequencing-based association studies of global populations. *Genet Epidemiol*. 2017;41:145-151.
12. Landrum MJ, Lee JM, Benson M, Brown G, Chao C, Chitipiralla S, Gu B, Hart J, Hoffman D, Hoover J, Jang W, Katz K, Ovetsky M, Riley G, Sethi A, Tully R, Villamarin-Salomon R,

Rubinstein W and Maglott DR. ClinVar: public archive of interpretations of clinically relevant variants. *Nucleic Acids Res*. 2016;44:D862-868.
13. Khera AV, Won HH, Peloso GM, Lawson KS, Bartz TM, Deng X, van Leeuwen EM, Natarajan P, Emdin CA, Bick AG, Morrison AC, Brody JA, Gupta N, Nomura A, Kessler T, Duga S, Bis JC, van Duijn CM, Cupples LA, Psaty B, Rader DJ, Danesh J, Schunkert H, McPherson R, Farrall M, Watkins H, Lander E, Wilson JG, Correa A, Boerwinkle E, Merlini PA, Ardissino D, Saleheen D, Gabriel S and Kathiresan S. Diagnostic Yield and Clinical Utility of Sequencing Familial Hypercholesterolemia Genes in Patients With Severe Hypercholesterolemia. *J Am Coll Cardiol*. 2016;67:2578-2589.