Supplementary Methods

Pinal-Fernandez I, *et al. Ann Rheum Dis* 2020; 79:1234 1235–1242 1243. doi: 10.1136/annrheumdis-2019-216599

1. Cultured human skeletal muscle cells

Normal human skeletal muscle myoblasts (HSMM; Lonza) were cultured according to the manufacturer's protocol. When 80% confluent, the cultures were induced to differentiate into myotubes by replacing the growth medium with differentiation medium (DMEM, 2% horse serum, and L-glutamine). Two plates of cells were collected for RNA extraction at 7 separate time points: immediately before differentiation and then daily for 6 days.

2. Mouse Muscle Injury

Muscle injury and regeneration were induced in mice using cardiotoxin (CTX) as previously described.[1] Briefly, 6 week-old C57BL/6 mice were unilaterally injured by intramuscular injection of 0.1 mL of 10 μM CTX into the tibialis anterior (TA) muscle. Injured TA muscles were harvested at days 3 (n=2), 5 (n=2), 7 (n=2), 10 (n=4), 14 (n=4), and 28 (n=3) post-injury. Contralateral (uninjured) TA muscles were also collected (n=9). Muscle tissue was snap-frozen and stored at -80 degrees Celsius.

Pinal-Fernandez I, *et al. Ann Rheum Dis* 2020; 79:1234 1235–1242 1243. doi: 10.1136/annrheumdis-2019-216599

3. Human muscle biopsy processing

a) Freezing of muscle tissues

Open muscle biopsies were placed in an aluminum foil envelope. 2-methylbutane (isopentane) was pre-chilled using liquid nitrogen and the aluminum foil envelopes were submerged in the isopentane for 15 seconds. After this, the samples were placed in cryovials at -80º for long-term storage. Samples collected at other institutions were shipped in dry ice to the NIH Muscle Disease Unit.

b) RNA extraction protocol

Required reagents

1. TRIzol (ThermoFisher # 15596026)
2. Chloroform
3. 100% isopropanol
4. 75% ethanol
5. 1.4 ceramic bead homogenizing tubes (VWR # 10032-358)
6. RNase-free water
7. 1.5 mL low-binding tubes
8. GlycoBlue (ThermoFisher #AM9515)

Pinal-Fernandez I, *et al. Ann Rheum Dis* 2020; 79:1234 1235–1242 1243. doi: 10.1136/annrheumdis-2019-216599

Step 0: Set-up reagents and equipment

1. Spray down centrifuge, then set at 4º.
2. Set heat block at 55ºC.
3. Set Precellys 24 homogenizer at 3x15'' and 6500rpm.
4. Obtain dry ice and ice.
5. Thaw GlycoBlue.

Step 1: Homogenization

1. Get TRIzol from 4º.
2. Get homogenization tubes with beads.
3. Label tubes and set in dry ice.
4. Get samples from -80ºC and place in dry ice.
5. Cut 1-2mm piece of the muscle biopsy with a #11 surgical blade.
6. Fill tubes in ice with 1mL TRIzol and immediately homogenize 3 x 15 minutes.

Step 2: Phase separation

1. Get chloroform, 100% isopropanol, 75% ethanol, and RNase-free water.
2. Wait for 5 minutes at room temperature.
3. Add 0.2 mL of chloroform.
4. Shake 15 seconds by hand.
5. Incubate 2-3 minutes at room temperature.
6. Centrifuge the sample at 12,000g for 15 minutes-4ºC.
7. Label new 1.5mL tubes for all 4 samples.

Pinal-Fernandez I, *et al. Ann Rheum Dis* 2020; 79:1234 1235–1242 1243. doi: 10.1136/annrheumdis-2019-216599

8. Remove the aqueous phase samples, being careful not to touch interphase.
9. Place aqueous phases into new tubes.
10. Store interphase and organic phases -80ºC.

Step 3: RNA Isolation

1. Add 1.5uL of GlycoBlue, flicker and short spin.
2. Add 0.5 mL of 100% isopropanol.
3. Incubate at room temperature for 10 minutes.
4. Centrifuge the sample at 12,000g for 10 minutes at 4ºC.

Step 4: RNA wash

1. Remove supernatant (leave 0.5 mm).
2. Wash pellet with 1mL of 75% ethanol.
3. Vortex sample briefly.
4. Centrifuge the sample at 7,500g for 5 minutes at 4ºC.
5. Discard the wash (with a pipette).
6. Dry RNA pellet for 5 minutes.

Step 5: RNA resuspension

1. Resuspend RNA pellet in 40µL RNase-free water.
2. Incubate on heat block 10 minutes.
3. Measure the quantity of RNA with NanoDrop.

Pinal-Fernandez I, *et al. Ann Rheum Dis* 2020; 79:1234 1235–1242 1243. doi: 10.1136/annrheumdis-2019-216599

Step 6: Evaluate RNA quality with 4200 TapeStation

1. Allow High Sensitivity RNA Sample buffer (5067- 5580) to equilibrate at room temperature for 30 minutes.
2. Thaw High Sensitivity RNA ladder (5067- 5581) and total RNA samples on ice.
3. Launch the Agilent 4200 TapeStation Controller Software and select RNA assay mode under “settings”.
4. Flick the High Sensitivity RNA ScreenTape device (5067- 5579) and load it into the 4200 TapeStation instrument.
5. Place loading tips (5067- 5598) into the Agilent 4200 TapeStation instrument.
6. Vortex reagents and spin down before use.
7. Prepare diluted Ladder solution by adding 10 µL RNase free water to the High Sensitivity RNA
8. Ladder vial and mix thoroughly. Pipette 1 µL High Sensitivity RNA Sample Buffer and 2 µL diluted High Sensitivity RNA Ladder at position A1 in a tube strip (401428).
9. For each sample, pipette 1 µL High Sensitivity RNA Sample Buffer and 2 µL RNA sample in a tube strip.
10. Cap tube strips with ladder or sample.
11. Mix liquids in sample and ladder vials using the IKA vortex at 2000 rpm for 1 min.
12. Spin down to position the sample and ladder at the bottom of the well plate and tube strip.

Pinal-Fernandez I, *et al. Ann Rheum Dis* 2020; 79:1234 1235–1242 1243. doi: 10.1136/annrheumdis-2019-216599

13. Samples and ladder denaturation:
    a. Heat samples and ladder to 72 °C (162 °F) for 3 min.
    b. Place samples and ladder on ice for 2 min.
    c. Spin down to position the samples and ladder at the bottom of the well plate and tube strip.
14. Sample Analysis:
    a. Load samples into the Agilent 4200 TapeStation instrument. Carefully remove caps of tube strips.
    b. Place the ladder in position A1 on tube strip holder in the 4200 TapeStation instrument.
    c. Select the required sample positions on the 4200 TapeStation Controller Software.
    d. Click “Start”.
    e. The Agilent Tapestation Analysis Software opens after the run and displays results.

c) Sample size requirements, RNA yield, and quality

- The frozen muscle biopsy specimens were placed on glass pre-cooled with dry ice and a 1-2mm section of muscle was removed with a #11 surgical blade.

- An average of 11ug (SD 12ug) of RNA was recovered from each muscle biopsy specimen. 65ng of the samples were used to prepare the RNA library using the

Pinal-Fernandez I, *et al. Ann Rheum Dis* 2020; 79:1234 1235–1242 1243. doi: 10.1136/annrheumdis-2019-216599

NeoPrepTM system according to the TruSeq Stranded mRNA Library Prep protocol (Illumina) and sequenced using the Illumina HiSeq 2500 or 3000.

- The integrity of the RNA was verified using a standard quality metric denominated RNA integrity number (RIN) value, showing a median value of 7 (interquartile range [IQR] 5.9–7.4) for the muscle biopsy samples.

Pinal-Fernandez I, *et al. Ann Rheum Dis* 2020; 79:1234 1235–1242 1243. doi: 10.1136/annrheumdis-2019-216599

4. Differential gene expression

We performed the differential expression between different subgroups using DESeq2 v.1.20.[2] DESeq2 performs an internal normalization where the geometric mean is calculated for each gene across all samples. The counts for a gene in each sample is then divided by this mean. The median of these ratios in a sample is the size factor for that sample. This procedure corrects for library size and RNA composition bias, which can arise for example when only a small number of genes are very highly expressed in one experiment condition but not in the other.

Additionally, DESeq2 automatically detects count outliers using Cooks' distance and removes these genes from the analysis. DESeq2 v.1.20 also performs independent filtering which maximizes the number of genes which will have a Benjamini and Hochberg-adjusted p–value less than a critical value set by default to 0.1; removing these genes with low counts improves the detection power by making the multiple testing adjustment of the p-values less severe. To speed up the computations we prefiltered genes with a total count across conditions below 10. Since these genes would have been excluded from the analysis afterward anyways, this did not influence the calculations at all.

DESeq2 uses shrinkage estimation for dispersions and fold changes. A dispersion value is estimated for each gene through a model fit procedure. Using these estimations, the package fits a negative binomial generalized linear model for each gene and uses the Wald test for significance testing. The Wald test P values

Pinal-Fernandez I, *et al. Ann Rheum Dis* 2020; 79:1234 1235–1242 1243. doi: 10.1136/annrheumdis-2019-216599

from the subset of genes that pass the independent filtering step are adjusted for multiple testing using the procedure of Benjamini and Hochberg.[3]

Loading the data, prefiltering and fitting the model was done with the following code:

```
library(DESeq2)
if (packageVersion("DESeq2") != "1.20.0") {
  stop("DESeq2 version is not 1.20.0, please use version 1.20.0")
}

#Set working directory
project_folder <- "/working_directory_path"
setwd(project_folder)

#Import count data and the sample covariates
countData <- as.matrix(read.csv("./anonymized_gene_counts.csv", row.names="gene_id"))
sample_covariates <- t(read.csv("./anonymized_gene_counts.csv", nrows=1, header = FALSE)[-1])
colnames(sample_covariates) <- "GROUP"

#Load data in DESeq2
dds <- DESeqDataSetFromMatrix(countData = countData, colData = sample_covariates, design= ~ GROUP)

#Prefiltering
dds <- dds[ rowSums(counts(dds)) >= 10, ]

#Fitting the model
deseq_fitted <- DESeq(dds, betaPrior=T)
```

To ensure the stability of the central tendency and dispersion values of each biological group between different sections of the study, the normalization process included the totality of the samples even if that specific comparison did not include some of those samples.

For example, the comparison between anti-Jo1 and normal biopsies was performed using the following code:

```
#Calculating the differential expression between anti-Jo1 and normal biopsies
deseq_results <- results(deseq_fitted, contrast=c("GROUP", "Jo1", "NT"))
write.csv(deseq_results[with(deseq_results, order(padj, pvalue)),], file=("./jo1_vs_nt.csv"))
```

We assigned equal weights to each autoantibody subgroups within DM and IMNM to avoid giving more importance to differentially expressed transcriptomic features of autoantibody subgroups with a higher number of biopsies at this stage of the analysis.

For example, to compare DM and IMNM we used the following code:

```
#Calculating the differential expression between DM and IMNM biopsies
resultsNames(deseq_fitted)
#'Intercept' 'GROUPHMGCR' 'GROUPIBM' 'GROUPJo1' 'GROUPMDA5' 'GROUPMi2' 'GROUPNT' 'GROUPNXP2' 'GROUPSRP'
'GROUPTIF1'
deseq_results <- results(deseq_fitted, contrast=c(0,-1/2,0,0,1/4,1/4,0,1/4,-1/2,1/4))
write.csv(deseq_results[with(deseq_results, order(padj, pvalue)),], file=("./dm_vs_imnm.csv"))
```

5. RNAseq-based classification

a) Import data and normalize FPKM

After importing the FPKM levels of all genes, we performed a logarithmic transformation and then normalized the data, maintaining the relative expression of the gene levels.

The rationale behind log-transforming the RNA expression values was to make variation similar across different orders of magnitude.[4]

As per the gene normalization, given that genes with low expression levels are more prone to technical bias in RNAseq,[5] the different genes were normalized so the relative expression of the gene levels would be preserved.

The following code was used to perform this step of the analysis:

```
import pandas as pd
import numpy as np

#Import data
df = pd.read_csv('anonymized_gene_fpkm.csv')

#Generate lists of samples
nt = df['GROUP'] == 'NT'
dm = df['GROUP'].isin(['Mi2', 'NXP2', 'TIF1', 'MDA5'])
asys = df['GROUP'] == 'Jo1'
imnm = df['GROUP'].isin(['HMGCR', 'SRP'])
ibm = df['GROUP'] == 'IBM'
groups = {'nt':nt, 'dm':dm, 'as':asys, 'ibm':ibm, 'imnm':imnm}

df = df.set_index('GROUP')

#Log-transform FPKM values
df = np.log2(df+1)

#Normalize FPKM maintaining relative expression of gene levels
df = (df-np.mean(df.values))/np.std(df.values)
```


Pinal-Fernandez I, *et al. Ann Rheum Dis* 2020; 79:1234 1235–1242 1243. doi: 10.1136/annrheumdis-2019-216599

b) Filter genes with low signal-to-noise ratio

To filter genes with low signal-to-noise ratio, we selected all the genes that were significantly different (with a cutoff of q-value <0.05) in each group compared to the rest. This was performed using the following lines of code:

```
#Filter genes with q-value>0.05
diff_expression_dfs = {}
sig_genes = []

for group in groups.keys():
    diff_expression = pd.read_csv('./' + group + '_vs_all.csv')
    diff_expression = diff_expression[diff_expression['padj'] < 0.05]
    sig_genes = sig_genes + list(diff_expression['gene'])
sig_genes = list(set(sig_genes))
sig_genes = [gene.replace('-', '_').upper() for gene in sig_genes]
df = df[sig_genes]
df.shape
```

c) Stratified cross-validation

We performed a stratified 3-fold-cross-validation of the samples included in the study. Stratification was used in order to ensure that there were enough samples in each cycle to build the models.

d) Model training

With the objective of showing that the information contained in the RNAseq has classificatory value in myositis, we tested a set of classificatory machine learning models using the default parameters (for example, for AdaBoost we used the default algorithm AdaBoost-SAMME)[6]. The rationale behind performing this screening was that there were no comprehensive studies in the field to guide our selection of the best model for this application.

Pinal-Fernandez I, *et al. Ann Rheum Dis* 2020; 79:1234 1235–1242 1243. doi: 10.1136/annrheumdis-2019-216599

The cross-validation and training of the models were performed with the following lines of code:

```
import logging
from sklearn.model_selection import StratifiedKFold
from sklearn import metrics
from sklearn.svm import LinearSVC, SVC
from sklearn.ensemble import RandomForestClassifier, AdaBoostClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.tree import DecisionTreeClassifier
from sklearn.neural_network import MLPClassifier
from sklearn.gaussian_process import GaussianProcessClassifier
from sklearn.discriminant_analysis import QuadraticDiscriminantAnalysis

logging.basicConfig(level=logging.INFO, filename="models_log.log")

np.random.seed(1)

classifiers = {
    "Linear SVM": LinearSVC(),
    "RBF SVM": SVC(),
    "Random Forest": RandomForestClassifier(),
    "Nearest Neighbors": KNeighborsClassifier(),
    "Gaussian Process": GaussianProcessClassifier(),
    "Decision Tree": DecisionTreeClassifier(),
    "Neural Net": MLPClassifier(),
    "AdaBoost": AdaBoostClassifier(),
    "Gaussian Naive Bayes": GaussianNB(),
    "QDA": QuadraticDiscriminantAnalysis()
    }

for model, clf in zip(classifiers.keys(), classifiers.values()):
    results=[]

    print(model)
    logging.info(model)

    for j in range(1000):
        results_set = []

        for i in groups.values():
            #Diagnostic value
            skf = StratifiedKFold(n_splits=3,shuffle=True)
            skf.get_n_splits(df, i)

            train_index, test_index = next(skf.split(df, nt))

            try:
                clf.fit(df.iloc[train_index], i.iloc[train_index])
                y_pred=clf.predict(df.iloc[test_index])
            except:
                pass

            results_set.append(metrics.accuracy_score(i.iloc[test_index], y_pred))

        results.append(np.array(results_set))
        logging.info(["{0:0.2f}".format(item) for item in results_set])

    mean = np.mean(results, axis=0)
    ci_lower = np.percentile(results, 2.5, axis=0)
    ci_upper = np.percentile(results, 97.5, axis=0)

    for n, i in enumerate(groups.keys()):
        print("   {0}: {2:0.1f} [{1:0.1f}-{3:0.1f}]".format(i, ci_lower[n]*100, mean[n]*100, ci_upper[n]*100))
```

e) Comparing the effect of including the gene-selection in the internal validation.

We decided to use all the genes that were significantly different (with a cutoff of q-value <0.05) in each group compared to the rest using all the samples. An alternative would have been to include the differentially expressed genes contained in the training set of each cycle. However, this approach was excessively computationally expensive. Nonetheless, to demonstrate the equivalency of these approaches, we modified our pipeline to train 100 cross-validation cycles using only the differentially expressed genes resulting from each training set. The performance of the models was equivalent using both methods (compare Table 2 with Supplementary Table 3).

f) Gene ranking based on the linear support vector machine classification

Once we determined that the linear SVM outperformed the rest of the models, we trained the whole dataset using this algorithm and then applied the recursive feature elimination technique [7] one gene at a time to sort the importance of the genes for each group of subjects.

Pinal-Fernandez I, *et al. Ann Rheum Dis* 2020; 79:1234 1235–1242 1243. doi: 10.1136/annrheumdis-2019-216599

The following lines of code were used to rank the genes for the different groups of subjects.

```
from sklearn.feature_selection import RFE

for name, dataset in zip(groups.keys(), groups.values()):
    print("Top 10 variables " + name)

    rfe = RFE(LinearSVC(), 1)
    rfe = rfe.fit(df, dataset)

    results = pd.DataFrame(sorted(zip(rfe.ranking_, df.columns)), columns = ['position', 'gene'])

    results.to_excel('../svm_gene_importance_' + name + '.xlsx', index=False)

    print(results.head())
```


Pinal-Fernandez I, *et al*. *Ann Rheum Dis* 2020; 79:1234 1235–1242 1243. doi: 10.1136/annrheumdis-2019-216599

References

1. Mammen AL, Casciola-Rosen LA, Hall JC, Christopher-Stine L, Corse AM, Rosen A. Expression of the dermatomyositis autoantigen Mi-2 in regenerating muscle. Arthritis Rheum. 2009 Dec; 60(12):3784-3793.
2. Love MI, Huber W, Anders S. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. Genome Biol. 2014; 15(12):550.
3. Benjamini Y, Drai D, Elmer G, Kafkafi N, Golani I. Controlling the false discovery rate in behavior genetics research. Behav Brain Res. 2001 Nov 1; 125(1-2):279-284.
4. Zwiener I, Frisch B, Binder H. Transforming RNA-Seq data to improve the performance of prognostic gene signatures. PLoS One. 2014; 9(1):e85150.
5. Bhargava V, Head SR, Ordoukhanian P, Mercola M, Subramaniam S. Technical variations in low-input RNA-seq methodologies. Sci Rep. 2014 Jan 14; 4:3678.
6. Zhu J, Zou H, Rosset S, Hastie T. Multi-class AdaBoost.
7. Guyon I, Weston J, Barnhill S, Vapnik V. Gene selection for cancer classification using support vector machines. Mach Learn. 2002; 46(1-3):389-422.