1 Programming

In this Homework, we continue our definition of vector spaces in ML:

1. First, we rewrite all our previous definitions to avoid using \texttt{if} by using pattern matching. This means that \texttt{scale}, \texttt{add} and \texttt{dot} need new definitions. (The function \texttt{magn} doesn’t need to be changed since it is defined without \texttt{if} anyway.) When you do this, you will need a few explicit type qualifications to ensure that the functions operate on vectors, rather than integer lists.

2. Then we will define new functions: \texttt{project}, \texttt{projectAll} and \texttt{inSpan} (see below).

All the functions (the re-written and the new) should be placed in a single file \texttt{vector2.sml}.

2 Notation

Mathematically we write a vector using a boldface lowercase letter:

\[
v = (v_1, v_2, \ldots, v_n)
\]

Then we can define our existing operations:

\texttt{scale}(x,v)

\[
xv = (xv_1, xv_2, \ldots, xv_n)
\]

\texttt{add}(v,w)

\[
v + w = (v_1 + w_1, v_2 + w_2, \ldots, v_n + w_n, w_{n+1}, \ldots, w_m)
\]

if \(m\), the length of \(w\) is greater than \(n\), the length of \(v\). If \(m = n\), we end at the last common sum. If \(m < n\), then \(v\)’s elements continue past the end of the common sum.

\texttt{dot}(v,w)

\[
v \cdot w = v_1w_1 + v_2w_2 + \ldots + v_pw_p
\]

where \(p = \min\{m, n\}\).

\texttt{magn}(v)

\[
|v| = \sqrt{v \cdot v}
\]
3 New Operations

Our new operations are defined as follows:

\[ P_{w}v = \frac{v \cdot w}{w \cdot w} \]

\[ P_{\{w_1, \ldots, w_p\}}v = P_{w_1}v + \ldots + P_{w_n}v \]

\[ v \in \text{span}(S) \iff |v - P_Sv| < \epsilon \]

where \( \epsilon \) is a suitably small number, and vector subtraction is defined using addition:

\[ v - w = v + (-1)w \]

In your program, you should define FUDGE as a small number and use it for \( \epsilon \):

\[ \text{val FUDGE} = 0.000001; \]

4 Example

- \text{val w1:vector} = [1.0];
- \text{val w1} = [1.0] : vector
- \text{val w2:vector} = [0.0,1.0,0.0];
- \text{val w2} = [0.0,1.0,0.0] : vector
- \text{val w3:vector} = [0.0,0.0,0.0,2.0];
- \text{val w3} = [0.0,0.0,0.0,2.0] : vector
- \text{val S} = [w1,w2,w3];
- \text{val S} = [[1.0],[0.0,1.0,0.0],[0.0,0.0,0.0,2.0]] : vector list
- \text{projectAll}(S,[1.0,2.0,3.0,4.0]);
- \text{val it} = [1.0,2.0,3.0,4.0] : real list
- \text{inSpan}(S,[1.0,2.0,3.0,4.0]);
- \text{val it} = false : bool
- \text{projectAll}(S,[1.0,2.0,2.0,3.0]);
- \text{val it} = [1.0,2.0,2.0,3.0] : real list
- \text{inSpan}(S,[1.0,2.0,2.0,3.0]);
- \text{val it} = true : bool

5 Overloading and Polymorphism

Do Exercises 3 and 4 of Chapter 8 (page 131).

6 Submitting Your Work

Leave vector2.sml in your AFS folder. Turn in the book exercises on paper at the beginning of lecture.