для каждого обучающего примера $\mathbf{x}_i$. Величина L описывается *кусочно-линейной функцией*, показанной на рис. 12.17, и называется *кусочно-линейной функцией потерь*. Обозначим $z_i = y_i(\sum_{j=1}^{d} w_j x_{ij} + b)$. Если z_i больше или равно 1, то значение L равно 0. Но для меньших значений z_i величина L линейно возрастает при убывании z_i.

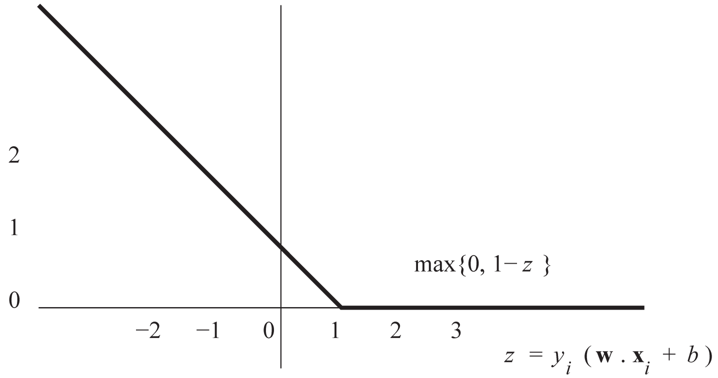

Рис. 12.17. Кусочно-линейная функция линейно убывает при $z \le 1$, а затем остается равной 0

Нам нужно будет брать частные производные функции $L(x_i, y_i)$ по каждому w_j, поэтому сразу отметим, что производная кусочно-линейной функции разрывна. Она равна $-y_i x_{ij}$ для $z_i < 1$ и 0 для $z_i > 1$. Следовательно, если $y_i = +1$ (т. е. i-й обучающий пример положителен), то

$$\frac{\partial L}{\partial w_j} = \text{ if } \sum_{j=1}^{d} w_j x_{ij} + b \ge 1 \text{ then } 0 \text{ else } -x_{ij}$$

С другой стороны, если $y_i = -1$ (т. е. i-й обучающий пример отрицателен), то

$$\frac{\partial L}{\partial w_j} = \text{ if } \sum_{j=1}^{d} w_j x_{ij} + b \le -1 \text{ then } 0 \text{ else } x_{ij}$$

Оба случая можно объединить в один, включив значение y_i:

$$\frac{\partial L}{\partial w_j} = \text{ if } y_i\left(\sum_{j=1}^{d} w_j x_{ij} + b\right) \ge 1 \text{ then } 0 \text{ else } -y_i x_{ij} \tag{12.5}$$

12.3.4. Нахождение решений в методе опорных векторов с помощью градиентного спуска

Общий подход к оптимизации функции (12.4) – использование квадратичного программирования. Но для больших данных предпочтение отдается другому подходу – *градиентному спуску*. Дело в том, что в отличие от квадратичного программирования, мы можем хранить данные на диске, а не целиком в оперативной па-